

TAMPEREEN AMMATTIKORKEAKOULU

Kone- ja tuotantotekniikan koulutusohjelma

Kone- ja laiteautomaatio

Tutkintotyö

Toni Kuusi

Simatic S7 -jatko-opintomateriaali

Työn ohjaaja

Työn teettäjä

Tampere 2007

Tekniikan lisensiaatti Olavi Kopponen

TAMK, valvojana laboratorioinsinööri Seppo Mäkelä

TAMPEREEN AMMATTIKORKEAKOULU

Kone- ja tuotantotekniikan koulutusohjelma

Kone- ja laiteautomaatio

Kuusi, Toni Simatic S7 –ohjekirja

Tutkintotyö 74 sivua

Työn ohjaaja Tekniikan lisensiaatti Olavi Kopponen

Työn teettäjä Tampereen ammattikorkeakoulu,
valvojana laboratorioinsinööri Seppo Mäkelä

Toukokuu 2007

Hakusanat operaatiot,logiikka,Simatic

TIIVISTELMÄ

Simatic manager S7 on automaatioalalla käytössä oleva tietokoneohjelma. Sen avulla suunnitellaan ja luodaan ohjelmia, jotka ladataan käyttöä varten logiikalle. Logiikka on ohjelmamuistina toimiva tietokone, jonka toiminta perustuu siihen ladattavaan ohjelmaan. Se ohjaa ladatun ohjelman avulla erilaisia toimilaitteita, esimerkiksi venttiileitä, moottoreita ja sylintereitä.

Tämä tutkintotyö tehtiin Tampereen ammattikorkeakoulun kone- ja tuotantotekniikan koulutusohjelman kone- ja laiteautomaation suuntautumisvaihtoehdon opetuskäyttöä varten. Aiheena oli laatia Simatic Manager S7 –ohjelman jatkokurssille opintomateriaali, joka antaa tietoa itse ohjelmasta ja sen käytöstä. Laatimisperiaatteena oli aikaansaada ohjeistus, josta lukijan on helppo löytää etsimänsä asiat. Ohjekirjasta löytyy tietoa ohjelmasta, sen osoitteista ja operaatioista. Se on myös ohjeistus ohjelmoimisesta, ohjelman testaamisesta sekä ohjelman lataamisesta.

TAMPERE POLYTECHNIC

Mechanical and Production Engineering

Machine Automation

Kuusi, Toni Simatic S7 – Manual

Engineering Thesis 74 pages

Thesis Supervisor Olavi Kopponen Lic. Tech.

Comissioning Company Tampere Polytechnic,
Supervisor Laboratory Engineer Seppo Mäkelä

May 2007

Keywords operation, logistics, Simatic

ABSTRACT

Simatic Manager S7 is a computer program for automation applications. It is used for creating and designing special programs, which can be downloaded to the logic. The logic itself is a computer, the function of which is based on this downloaded program. Utilizing the downloaded program, the logic controls various instruments and components, for example valves, motors and cylinders.

This engineer research report was made for the department of Machine Automation for where it can be used as a part of the education. The purpose of this report was to devise a manual that informs the reader how to use the program. The topic was to compile a learning material for the extension course of the Simatic Manager S7, that would give information about the program. The manual was designed to give the basic information about the program that would be easy for the reader to find. It also contains information about the program and its addresses and operations as well as instruction for the programming and the testing of the program.

ALKUSANAT

Tahdon kiittää Tampereen ammattikorkeakoulun kone- ja laiteautomaation suuntautumisvaihtoehtoa haastavasta ja mielenkiintoisesta tutkintotyöaiheesta. Opintomateriaalin tekeminen on ollut antoisaa ja mielenkiintoista. Tieto tutkintotyöni tarpeellisuudesta on osaltaan innostanut minua työnteon aikana. Erityisesti haluan kiittää työni valvojaa, laboratorioinsinööri Seppo Mäkelää, joka yhteistyöllään on avustanut minua koko tutkintotyöprosessin ajan ja antanut käyttööni tarvittavan materiaalin.

Tampereella 5.5.2007

Toni Kuusi

1 JOHDANTO	7
2 PROFIBUS DP	8
2.1 Yleistä Profibus-väylästä	8
2.2 Profibus DP:n ominaisuuksia ovat:	8
2.3 Tyypillisen Profibus DP järjestelmän osat ovat:	8
2.4 Hajautettu I/O SIEMENS ET 200M:n ominaisuuksia ovat:	9
2.5 Profibus-DP Monomaster-järjestelmä	9
2.6 Järjestelmän pystyttäminen	9
2.7 Asennus	10
2.8 Väyläprojektin luonti	11
2.9 CPU-tyypin valinta Profibus laitteistossa	11
2.10 Projektin nimi	13
2.11 Projektin rakenne	14
2.12 I/O Konfigurointi	15
2.13 I/O Moduulien liittäminen väylään	18
2.14 Väylämoduulien osoitteiden asettaminen	23
2.15 Clock Memoryn asettaminen	25
2.16 Konfiguroinnin lataaminen CPU:n muistiin	26
3 ASI-VÄYLÄN OHJE	30
3.1 Väylämoduulien alustus	30
3.2 Järjestelmän pystyttäminen	31
3.3 Asennus	33
3.4 Projektin luominen	34
3.5 Osoitteet	35
3.6 Ohjelmointi	37
4 TOIMINTAYKSIKÖT (Function Block, FB)	39
4.1 Ohjelmarakenne	39
4.2 Toimintayksiköt (FB)	39
4.3 Toimintayksikön luonti	40
4.4 S7-yksikön lisäys	41
5 KESKEYTYKSET	43
6 DATATYYPIN MUUNNOS	44
6.1 Datatyyppin muunnos operaatiot	44
6.2 Operaation valinat Simatic Manager -ohjelmassa	44
6.3 Esimerkki	49
7 VERTAILIJAT(Comparator)	50
7.1 Kokonaislukujen vertailu (16 bittiä)	50
7.2 Operaation valinat Simatic Manager -ohjelmassa	51
7.3 Esimerkki	52
7.4 Kokonaislukujen vertailu (32 bittiä)	52
7.5 Liukulukujen vertailu	53
7.6 Esimerkki	53
8 PERUSLASKENTAOPERAATIOIOT	54
8.1 Yleiskuva kokonaislukutoiminnoista	54
8.2 Operaation valinat Simatic Manager -ohjelmassa	54
8.3 Esimerkki 1	57
8.4 Esimerkki 2	57
9 SIIRTOKÄSKY (MOVE)	58
9.1 Kuvaus	58
9.2 Käskyjen valinta Simatic Manager –ohjelmassa	58
9.3 Esimerkki	59

10 YKSIKÖIDEN PARAMETROINTI	60
10.1 Yleistä.....	60
10.2 Parametrointi.....	60
10.3 Parametri.....	60
10.4 Staattiset arvot.....	61
10.5 Hetkelliset arvot	62
10.6 Paikallisosoitteet	62
11 Analoginen ohjelma	65
11.1 Automaatiolaitteen osoitteet	65
11.2 Analogiaosoitteet.....	65
11.3 Analogiaosoitteet S7-400.....	67
11.4 Liukuhihna esimerkki.....	67
12 DB -yksikön alustus	69
13 TULOKSET.....	72
14 TULOSTEN TARKASTELU	73
LÄHTEET	74

1 JOHDANTO

Simatic Manager S7 –ohjelma on automaatioalalla käytettävä ohjelma, jonka avulla voidaan luoda graafisia projekteja ja ladata ne logiikkaan. Ohjelma on laajalti suunnittelukäytössä eri automaatioyrityksissä sekä opetuskäytössä tekniikan alalla. Myös Tampereen ammattikorkeakoulun kone- ja laiteautomaation suuntautumisvaihtoehto on ottanut ohjelman opetuskäyttöön aiemmin käytetyn automaatio-ohjelman rinnalle.

Kuten muut kehittyneet automaatio-ohjelmat, Simatic Manager S7 on monipuolinen ohjelma ja vaatii käyttäjältään perehtymistä ohjelmaan. Tampereen ammattikorkeakoulussa oppilailla on käytössään ohjelmamanuaaleja ja paljon muuta tietoa ohjelmasta. Ongelmana on se, että tietoa on paljon ja se on hajallaan eri lähteissä. Jotta ohjelman oppiminen ja opettaminen koulussa helpottuisi, oli tietoa yhdistettävä.

Tämän tutkintotyön tavoitteena oli laatia ohjelman jatkokurssin asiat sisältävä ohjekirja sekä luoda tiivis kokonaisuus sen sisältämistä toiminnoista. Opetuksen kannalta tärkeä tieto oli eroteltava muusta tietomäärästä ja koottava yhteen. Ohjekirjan aiheenrajaus ja jäsentely on muodostettu työn teettäjän kanssa yhteistyössä. Tavoitteena oli tehdä ohjekirjasta mahdollisimman kattava ja johdonmukainen.

2 PROFIBUS DP

2.1 Yleistä Profibus-väylästä

Riippumattomat markkinatutkimukset vahvistavat PROFIBUS-väylän markkinajohtajuuden Euroopassa ja suurimman kasvun maailmanlaajuisesti. Asennuskanta PROFIBUS-väylällä on yli 1 500 000 laitetta 1997 vuotuisen kasvun ollessa 20 – 30 %./1/

1997 ARC markkinatutkimuksen mukaan kenttäväylien maailman markkinat kenttäväylille on yli 2.0 miljoonaa solmua, joista yli 50 % käytetään Euroopan alueella. 1996 Consultic -markkinatutkimuksen mukaan, joka keskittyy Saksan ja Euroopan markkinoille, PROFIBUS- väylän markkinaosuus Euroopassa on noin 41%./1/

2.2 Profibus DP:n ominaisuuksia ovat:

1. suunniteltu korvaamaan PLC:n kallis I/O – liityntä
2. erittäin nopea, siirtää 1 kByte input- ja output -dataa alle 2 ms:ssa
3. tehokkaat työkalut helppoon konfigurointiin ja ylläpitoon
4. johtavien PLC -toimittajien tukema
5. laaja tuotevalikoima: PLC, PC, I/O, käytöt, venttiilit, enkooderit
6. mono - ja multimaster verkot
7. 244 tavua tulo- ja lähtödataa asemaa kohti./1/

2.3 Tyypillisen Profibus DP järjestelmän osat ovat:

1. yksi PLC/PC Master-laitteena
2. erityyppisiä slavelaitteita
3. digitaali tai analogi I/O
4. AC- tai DC -käytöt
5. magneettiset tai pneumaattiset venttiilit
6. ihminen – kone -liityntä (HMI)./1/

2.4 Hajautettu I/O SIEMENS ET 200M:n ominaisuuksia ovat:

1. hajautus modulaarisesti
2. kaikki S7-300 korttiyksiköt käytettävissä
3. ei korttipaikkasääntöjä
4. korttiyksikköjen diagnostiikka
5. suojausluokka IP 20
6. tiedonsiirtonopeus max. 12 Mbit/s
7. kiinteä johdotus
8. 1-johdintekniikka (optiona 3-johdintekniikka)
9. aktiivinen pohjamoduuli: korttiyksiköiden vaihto mahdollista myös jännitteisenä./1/

2.5 Profibus-DP Monomaster-järjestelmä

Profibus DP –väylää käytetään pääasiallisesti yhden Master-laitteen väylänä. Tyypillisesti väylää käytetään logiikan hajautetun I/O:n toteutuksessa. Profibus-DP monomaster-järjestelmässä voi olla 1-125:een ”slavea”. Yhden Master-laitteen järjestelmillä saavutetaan erittäin lyhyt väylän kiertoaika. Nopeimmillaan 1 kt/s I/O-tietoa päivittyy 2 ms:n väyläjaksoissa./1/

2.6 Järjestelmän pystyttäminen

Seuraavat laitteet tarvitaan järjestelmään:

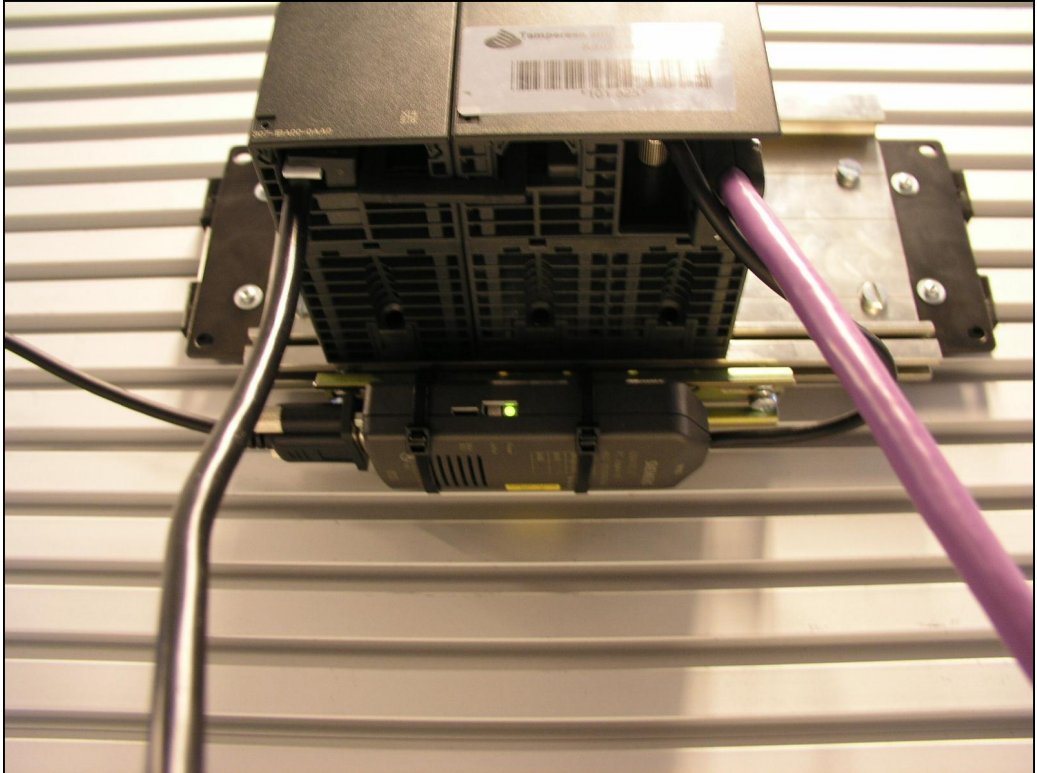
1. Simatic S7-315DP -logiikkayksikkö (PLC) (kaappi PLC 2, hylly PLC 2-7 tai PLC 2-6) (kuva 2)
2. PC, johon asennettu Simatic Manager –ohjelmisto
3. virtalähde (24 V DC) ja virtajohtoja (M1-15)
4. Siemens adapteri logiikan ja PC:n välille
5. hajautettu I/O ET 200M IM 153-2 (kaappi PLC 2, hylly PLC 2-8 tai PLC 2-9) (kuva 1)
6. hajautettu I/O ET 200B (kaappi PLC 2, hylly PLC 2-10 tai PLC 2-11)
7. väyläkaapeli (kaappi PLC 2).



Kuva 1 Hajautettu I/O ET 200M IM:n paikka luokassa

2.7 Asennus

Hajautetun tulo- ja lähtöyksikön (input and output modules, I/O) asennus kannattaa suorittaa vaiheittain seuraavan menettelyn mukaisesti. Kiinnitä logiikka ja hajautettu I/O Boschin asennuspöytään. Aseta väyläkaapelin päissä olevissa liittimissä päätevastukset ”on” - asentoon ja kaapelin keskimmäisessä liittimessä päätevastus ”off” - asentoon. Kytke logiikkaan virtajohtimet. Logiikka on toiminnassa, kun DC -ledi palaa. Kytke jännitteet myös I/O -moduuleihin. Huom. ET 200B 24 V. Kytke logiikkaan toinen väyläjohton pää. Kiinnitä väyläjohto sille varattuun paikkaan I/O -moduuleissa. Toisen pää voit kytkeä kummalle tahansa I/O -moduulille. Kytke adapteri siten, että MPI puoli (ilmoitettu adapterin kannessa) kiinnitetään logiikkaan ja RS232 puoli PC:n sarjaliittimeen. Valitse adapterin nopeudeksi 19.2 kb/s (kytkin adapterin sivussa). Käynnistä PC ja avaa Simatic Manager -ohjelma./1/



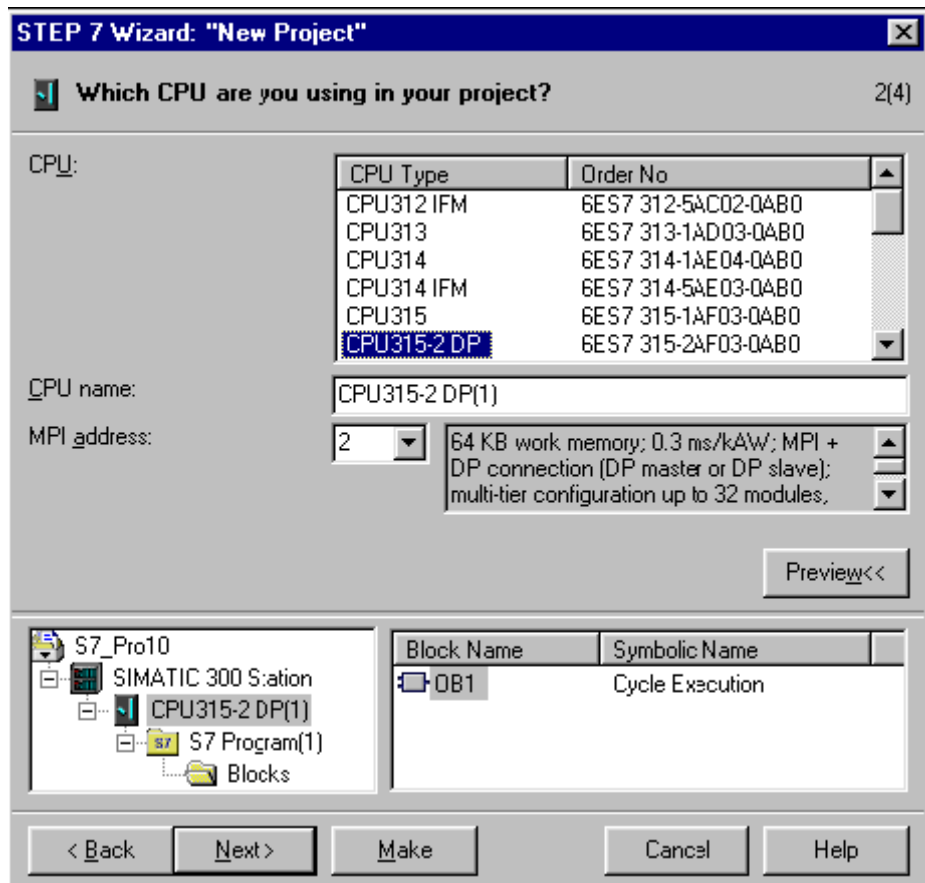
Kuva 2 Simatic S7-315DP -logiikkayksikkö

2.8 Väyläprojektin luonti

Esiasetukset suoritetaan STEP 7 -ohjeen mukaisesti aina kohtaan 1.3 saakka./1/

2.9 CPU-tyyppin valinta Profibus laitteistossa

Seuraavassa **Wizard** ikkunassa suoritetaan CPU-tyyppin valinta (kuva 3). Ikkunasta valitaan **CPU315-2DP** . MPI adress -numeroksi valitsemme 2 ja tämän jälkeen ”klikataan” **Next>** painiketta./1/

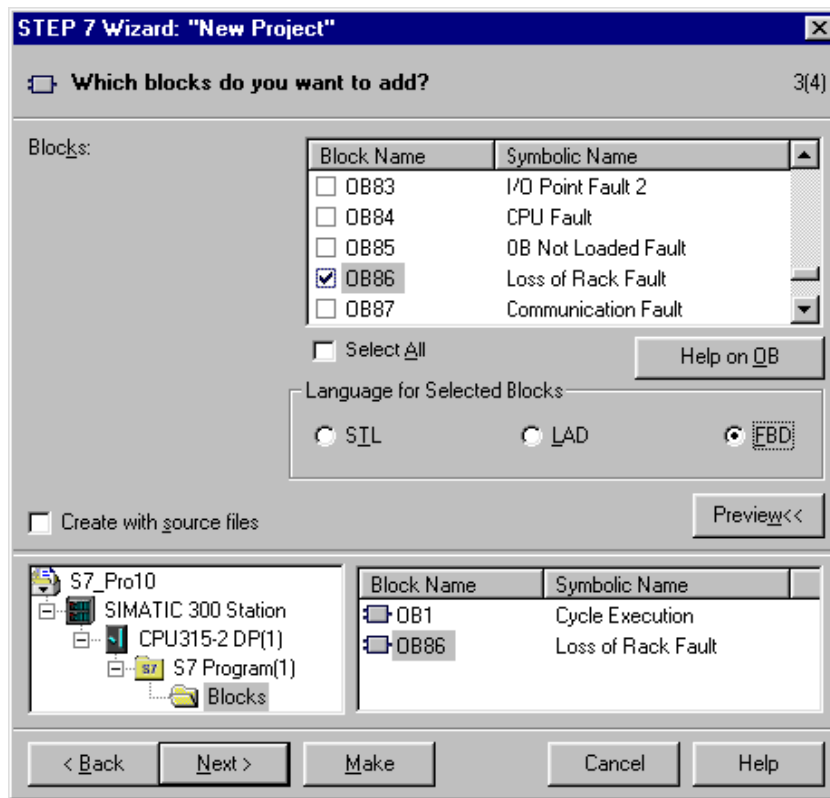


Kuva 3 CPU-tyypin valinta /1/

CPU valinnan jälkeen Wizard -ikkunasta valitaan OB:t (organisaatioblokit).

Peruskurssilla käytettiin ainoastaan OB1:tä. Nyt on välttämätöntä ottaa käyttöön myös OB 86, joten valitaan valintaruudukosta myös **OB86**.

Esitysmuodoksi (Language) valitaan *FBD* (kuva 4). Tästä edetään jälleen *Next>* -painikkeella seuraavaan ikkunaan./1/



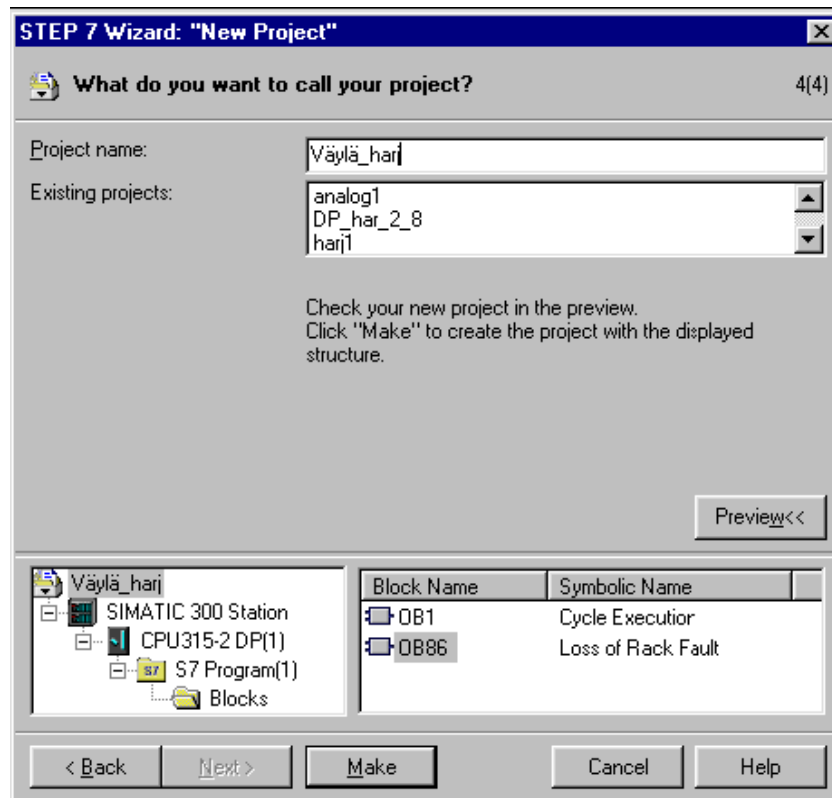
Kuva 4 Esitysmuodon valinta /1/

2.10 Projektin nimi

Viimeisessä Wizard-ikkunassa määritellään projektille nimi (Kuva 5).

Harjoitusohjelmissa käytetään nimenä henkilön etunimeä. Jos samalla nimellä on useampia ohjelmia, annetaan ohjelmalle lisäksi järjestysnumero esim.

Kalle_Ivöylä. Pakolliset esiasetukset on nyt suoritettu ja ”klikataan” **Make** -painiketta./1/

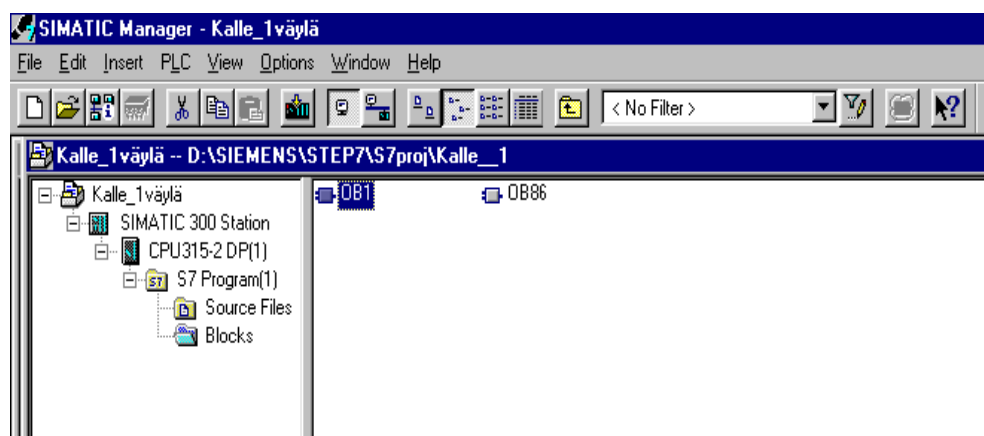


Kuva 5 Työn nimeäminen /1/

2.11 Projektin rakenne

Projektin muodostuksen jälkeen avautuu näkymä, jossa on projektin rakenne (Kuva 6). Ensimmäisellä tasolla sijaitsee projektin nimi **Kalle_1väylä**.

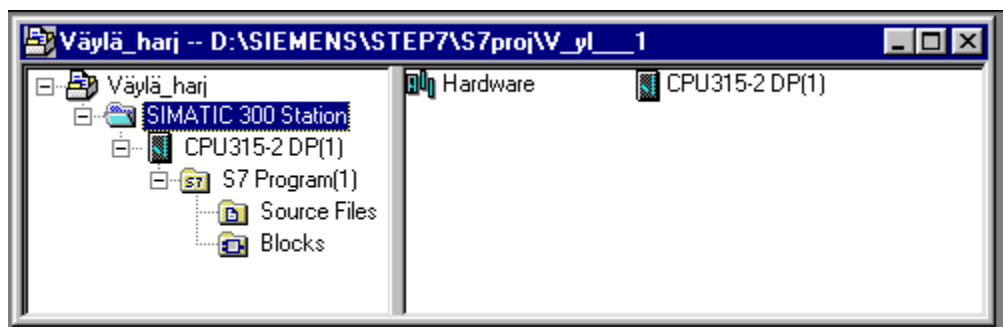
Projektiin valittu Station on meidän tapauksessamme **SIMATIC 300** . CPU, jota käytämme, on CPU315-2DP. Alimmalle tasolle muodostuu S7 Program -kansio, jossa PLC:n ohjelmablokit ja lähdekoodikansio sijaitsevat./1/



Kuva 6 Projektin rakenne /1/

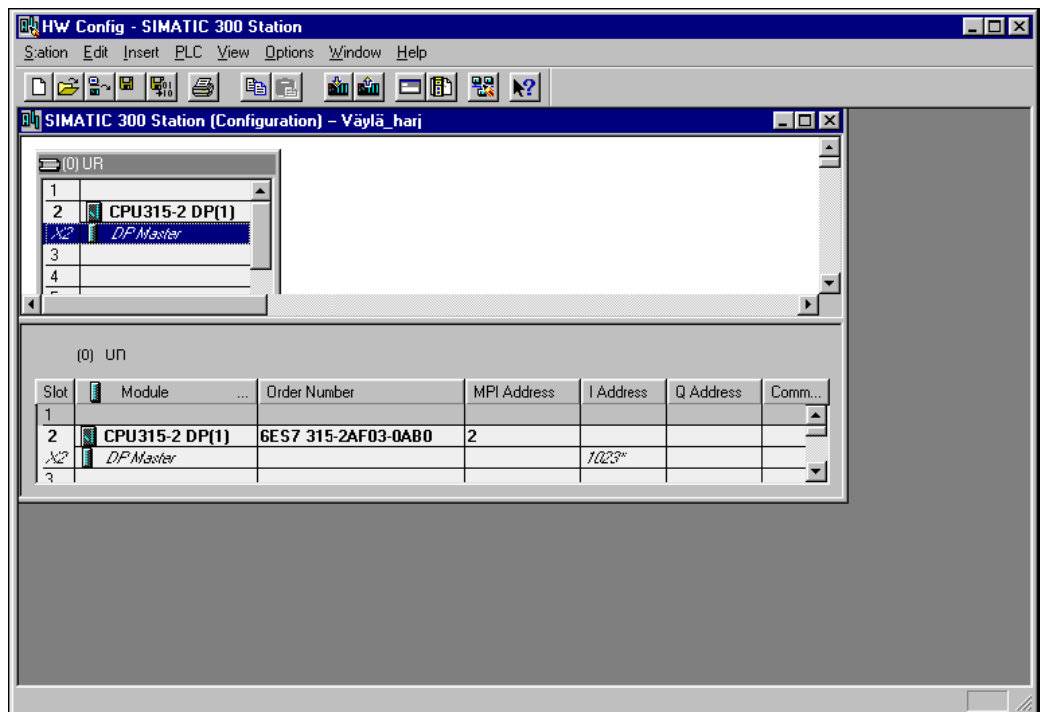
2.12 I/O Konfigurointi

Profibusväylän konfigurointi aloitetaan projektin rakennetta esittävästä kehikosta (Kuva 6). ”Kaksoisklikataan” *SIMATIC 300 Station* -ikonia, jolloin saadaan näkyviin kuvan 7 osoittama valintakehikko. Oikeanpuoleiseen kehikkoon tulee näkyviin *Hardware* ja *CPU315-2DP* -ikonit, näistä valitaan *Hardware ikoni* ”kaksoisklikkaamalla”. /1/



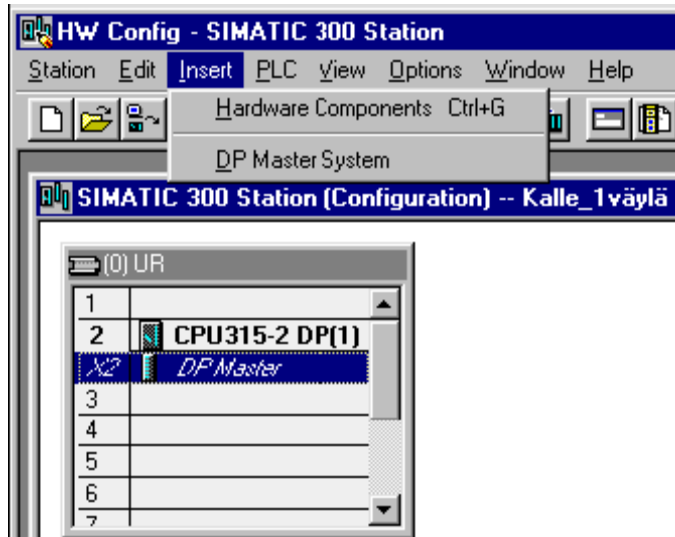
Kuva 7 Hardware ikonin valinta /1/

Hardware -ikonista avautuu kuvan 8 mukainen näkymä.



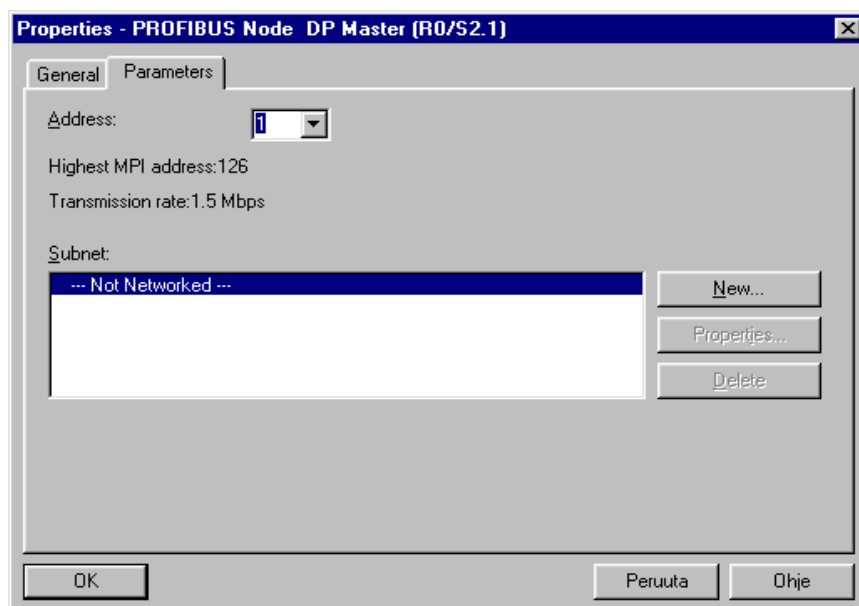
Kuva 8 Hardware -ikonista avautuva ikkuna /1/

(0)UR -kehykseen luodaan Profibus-väylän HW-konfigurointi. Ensimmäiseksi aktivoidaan (0)UR -kehyksen *DP Master rivi* (Kuva 9) ja *HW Config* ikkunan verhovalikosta valitaan *Insert* -painike, jolloin aukeaa verhovalikko. Valikon alimmalta riviltä ”klikataan” *DP Master System* -painiketta./1/

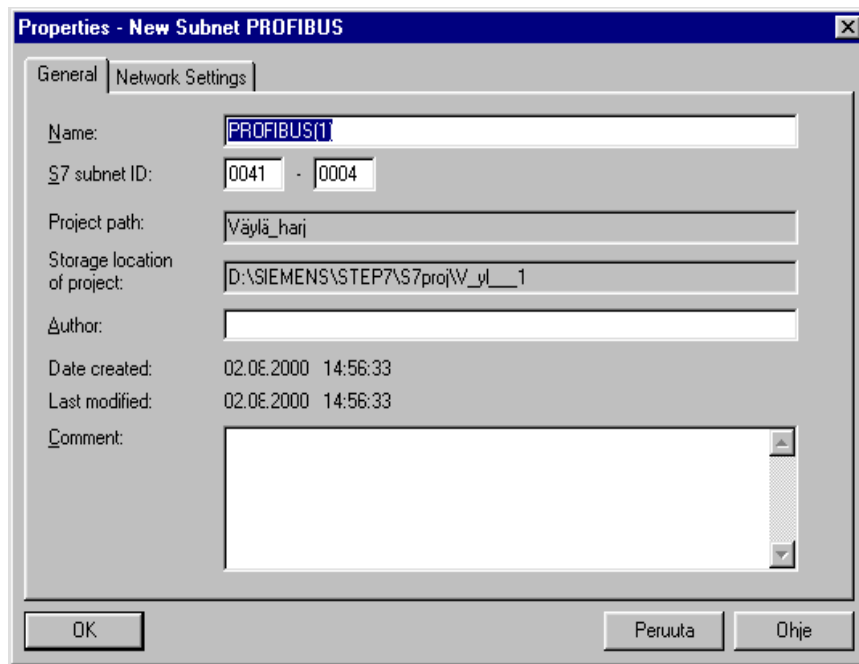


Kuva 9 DP Master Systemin valinta /1/

DP Master System valinnan jälkeen avautuu kuvan 10 mukainen PROFIBUS Node DP Master -ikkuna, josta address: asetukseksi valitaan *1* ja ”klikataan” *New* -painiketta, jolloin avautuu New Subnet PROFIBUS ikkuna (Kuva 11). /1/

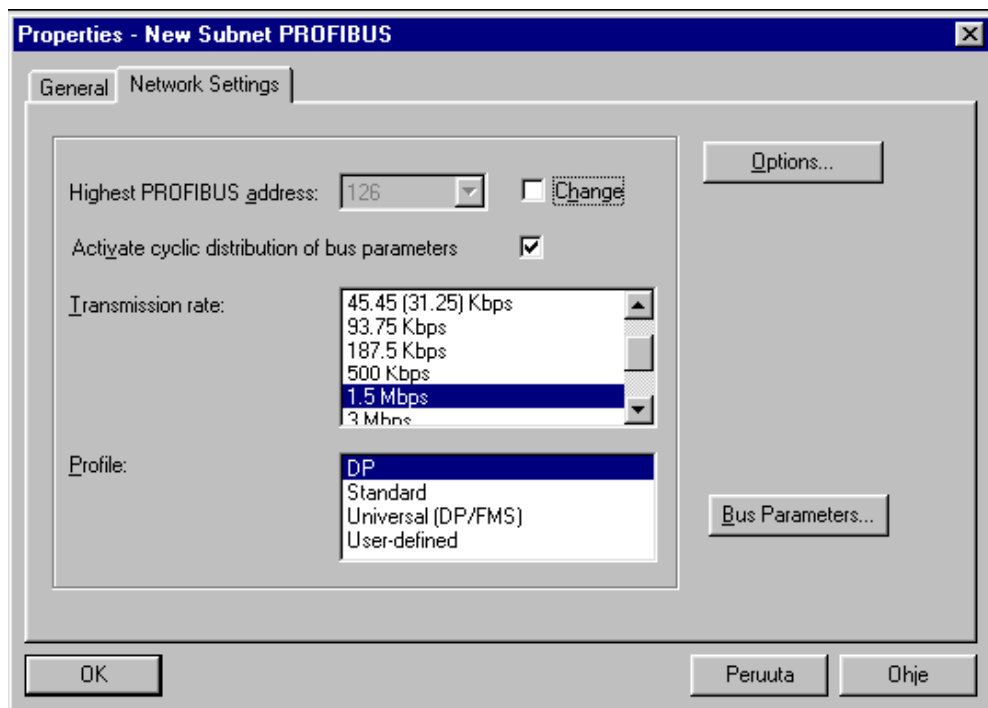


Kuva 10 Osoitteen valinta /1/



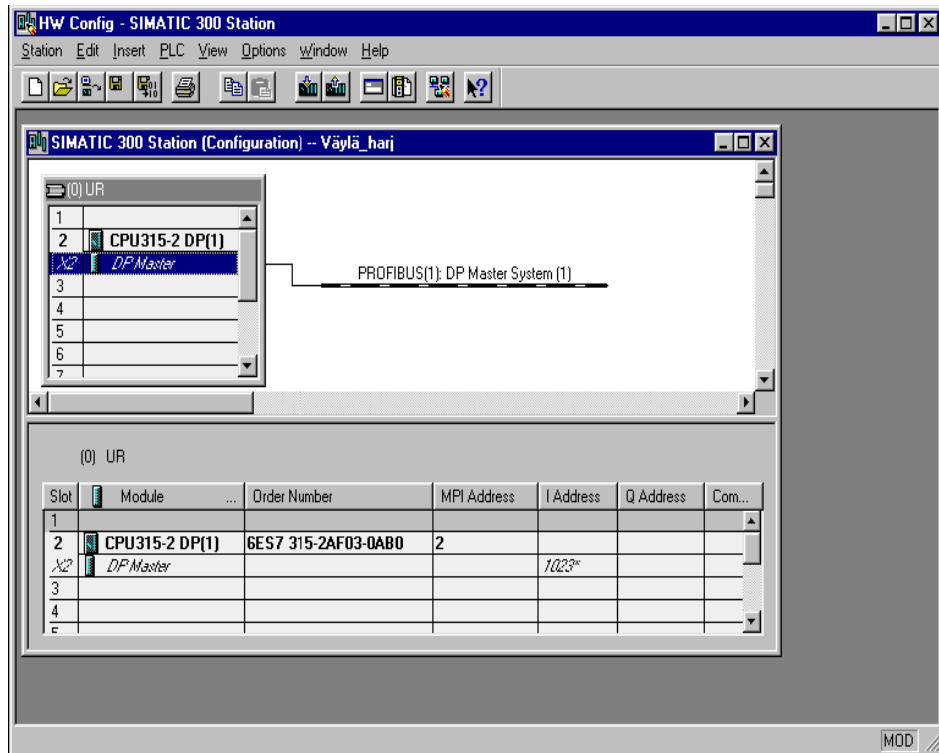
Kuva 11 New Subnet PROFIBUS –ikkuna /1/

General- välilehden asetuksiin ei tarvitse tehdä muutoksia, mutta *Network Settings* -sivulla (Kuva 12) tarkastetaan Transmission rate: nopeus, jonka pitää olla sama kuin adapterilla 19.2 kb/s ja Profile: asetukseksi asetetaan *DP*. /1/



Kuva 12 New Subnet PROFIBUS asetukset /1/

Options ja *Bus Parameters* -painikkeiden alla olevia oletusarvoja ei tarvitse muuttaa. Network Settings -sivun asetuksien jälkeen ”klikataan” OK, ja ohjelma palaa takaisin *PROFIBUS Node DP Master* -kehikkoon, jossa myös ”klikataan” OK -painiketta ja nyt aukeaa uudestaan HW Config -ikkuna (Kuva 13)./1/



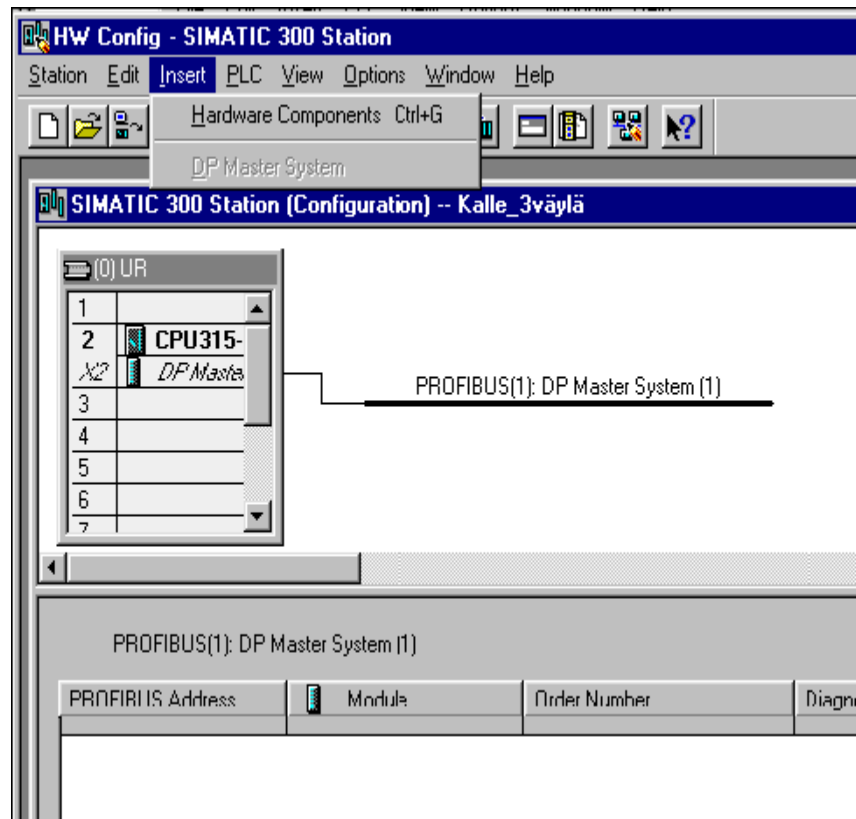
Kuva 13 DP Master System(1) -kaapeli /1/

Näiden toimenpiteiden jälkeen (0)UR -kehyksen DP Master riville on muodostunut PROFIBUS(1):DP Master System(1) kaapelia. Tähän väyläkaapeliin noudetaan kaikki väylään liittyvä komponentit. Tässä vaiheessa pitää tietää mitä komponentteja väylään liitetään. Mikäli kaikki osat ovat SIEMENS SIMATIC –komponentteja, löytyvät ne STEP 7 -ohjelmointiohjelman kirjastosta./1/

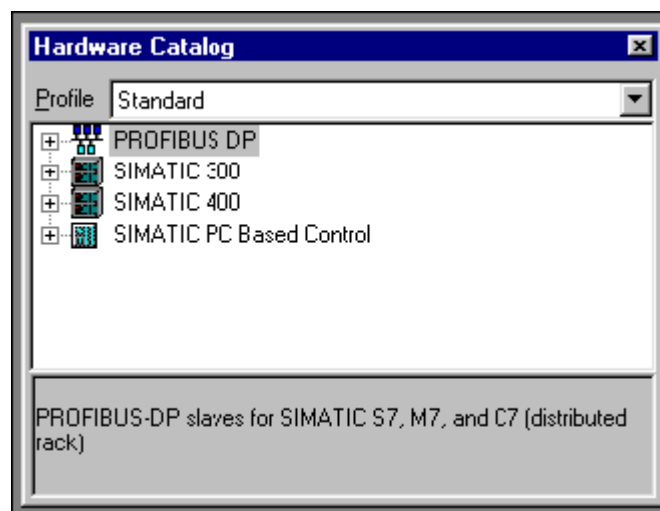
2.13 I/O Moduulien liittäminen väylään

Komponenttien noutaminen väylään aloitetaan aktivoimalla ”kerta-klikkauksella” väyläkaapeli, jolloin se muuttuu koko pituudeltaan mustaksi. Tämän jälkeen HW Config -kehikossa avataan Insert -verhovalikko (Kuva 14)

ja siitä valitaan *Hardware Components* -painike, jolloin Hardware Catalog avautuu näyttöruudulle (kuva 15)./1/

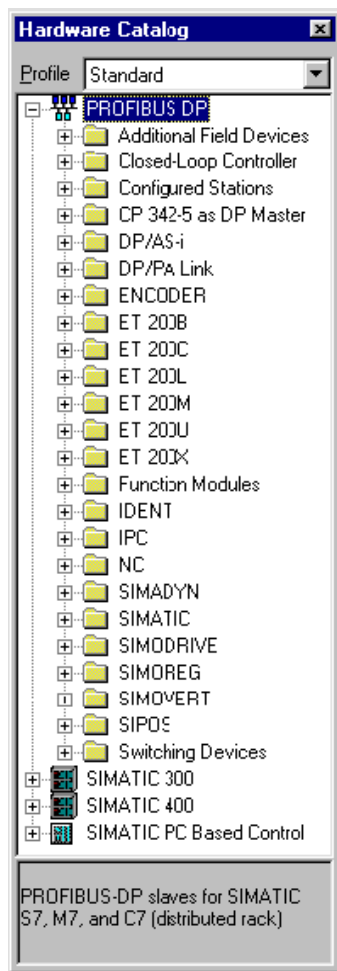


Kuva 14 Hardware Components -valinta /1/

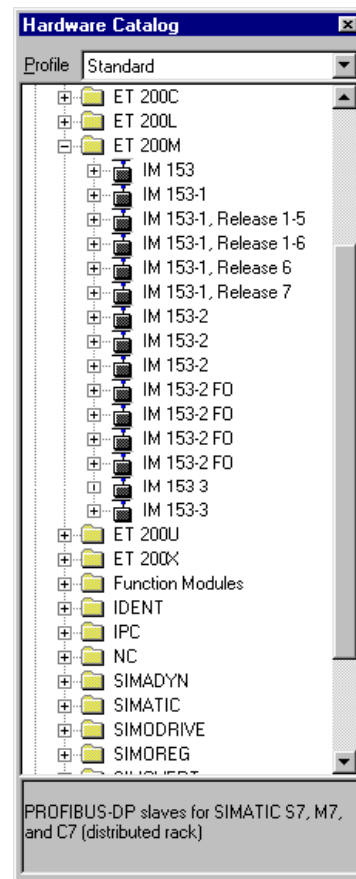


Kuva 15 PROFIBUS DP -kansion valinta /1/

Hardware Catalog -valintakehikosta avataan PROFIBUS DP -kansio. Se sisältää kaikki ne myynnissä olevat komponentit, jotka ovat olleet markkinoilla, kun ohjelmointiohjelma on päivitetty (Kuvat 16 ja 17)./1/



Kuva 16 Komponenttilista /1/

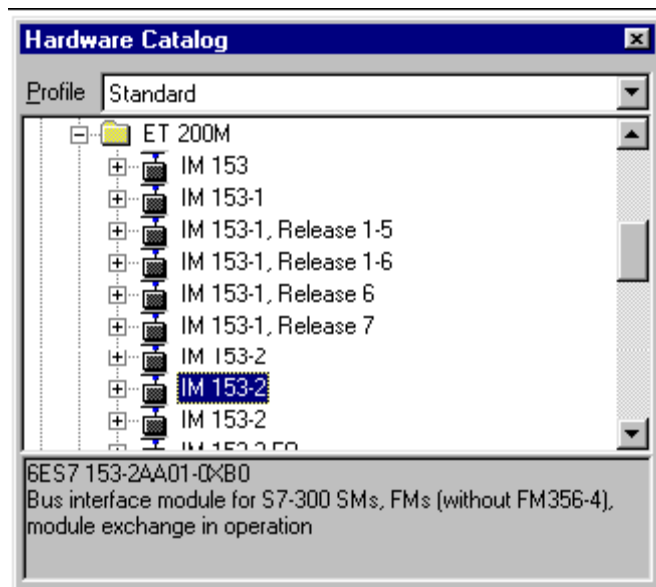


Kuva 17 Komponenttilista /1/

Kohdassa ”Järjestelmän pystyttäminen” on lueteltu ne I/O moduulit, jotka liitetään väylään. Jokaisen liitettävän osan tuotenumero pitää tietää, että oikeat komponentit voidaan valita luettelosta. Harjoitustyössä liitettävien slave komponenttien tuotenumerot ovat:

1. ET 200M IM 153-2
2. 153-2AA01-0XB0 (ET 200M perusmoduuli)
3. 322-1BH01-0AA0 (16 Output)
4. 321-1bh01-0AA0 (16 Input)
5. ET 200B
6. 133-0BH0-0XB0 (8 Input ja 8 Output). /1/

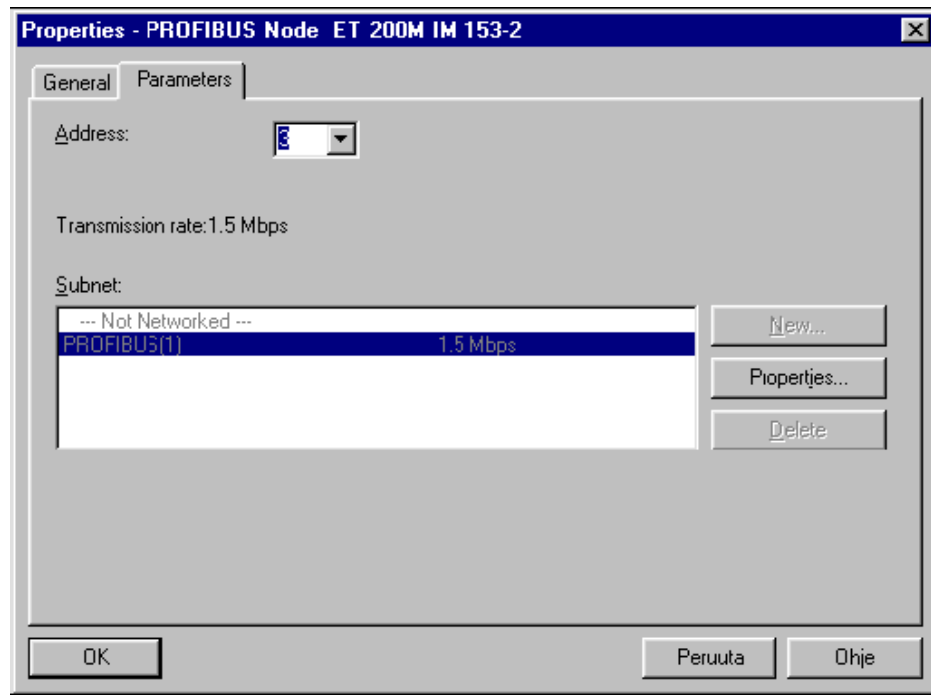
Ensimmäisenä väylään liitetään ET 200 M -perusmoduuli (Kuva 18).



Kuva 18 ET 200 M -perusmoduulin valinta /1/

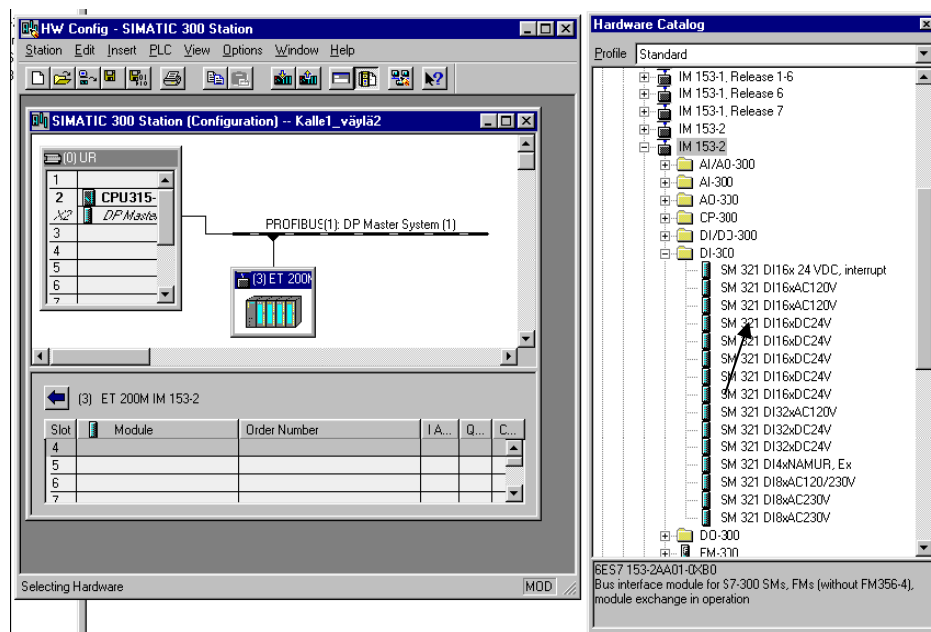
Liitettävän moduulin tyyppi on ET 200 M, joten valitaan aktiiviseksi kansio *ET200M*. Kyseisestä kansiosta löytyy kolme alikansiota, jotka omaavat tunnuksen 153-2. Oikean vaihtoehdon osaamme valita tarkkailemalla valintakehikon alaosassa olevaa harmaata aluetta, jossa näkyy aktiivisena olevan komponentin tuotenumero. Valitsemme keskimmäisen vaihtoehdon, koska siinä on etsityn komponentin tuotenumero (153- 2AA01-0XB0).

”Kaksoisklikkaamalla” *IM 153-2*:ta avautuu kuvan 19 mukainen valintakehys, jossa asetetaan ns. väyläosoite. Etenemme osoitteissa numerojärjestyksessä, ja seuraavana vuorossa on 3 (DP Master on 1 ja CPUon 2). /1/



Kuva 19 Väyläosoitteen asetus /1/

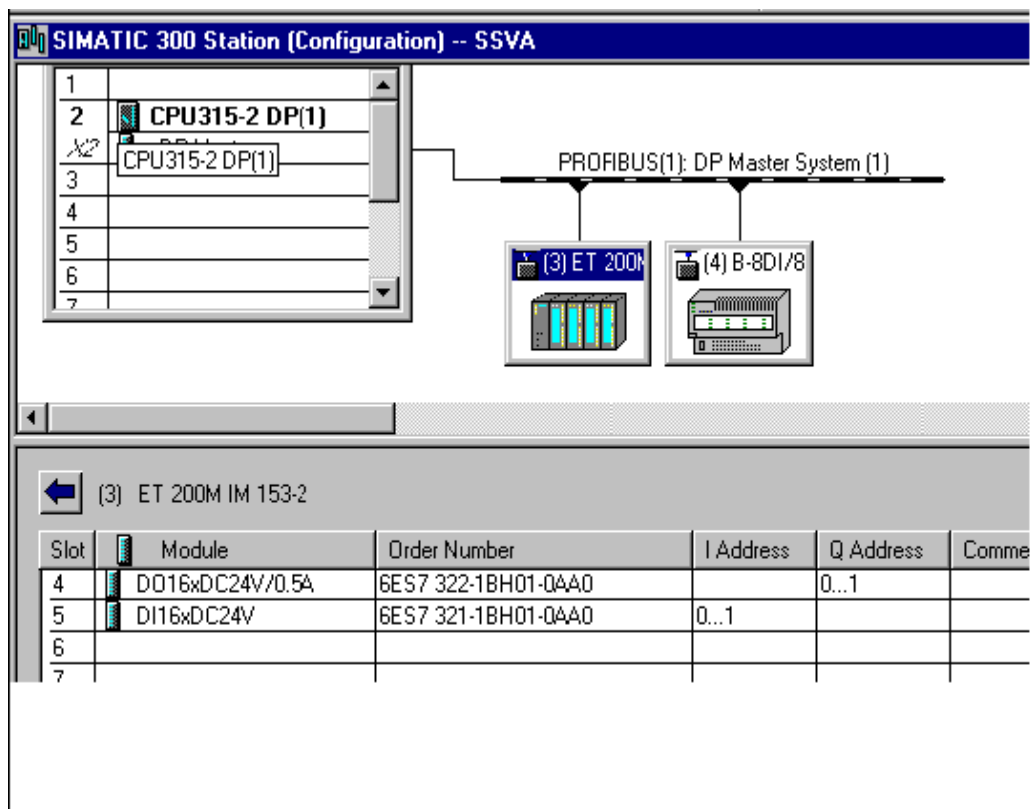
Valitsemalla kuvan 19 mukaisesta kehikosta OK painike siirtyy valittu yksikkö väyläkaapeliin (kuva 20).



Kuva 20 Valittu yksikkö siirtynyt väyläkaapeliin /1/

IM 153-2 perusmoduuliin haetaan vielä I/O yksiköt, jotka löytyvät, kun klikkaamme alikansion + merkkiä (Kuva 20). Nyt aukesi hakemisto, jossa on 9 alikansiota, sisältäen IM153-2 moduulin liitäntäyksiköitä. Valitsimme ensin

Input-moduulin, joka löytyy kansiota DO-300. Moduulin haku tapahtuu saman periaatteen mukaan kuin perusmoduulin IM 153-2 haku. Jotta DO-moduuli saadaan tuotua paikalleen, pitää väylällä olevan ET200-M moduulin olla aktiivinen, sekä saman kehikon alaosassa olevasta luettelosta aktivoidaan ensimmäinen vapaa rivi, joka tässä tapauksessa on rivi 4. Input-moduulin löydämme kansiota DO-300. Kun kaikki tarvittavat väylämoduulit on saatu väyläkaapeliin liitettyä, näyttää konfigurointi kuvan 21 mukaiselta. /1/

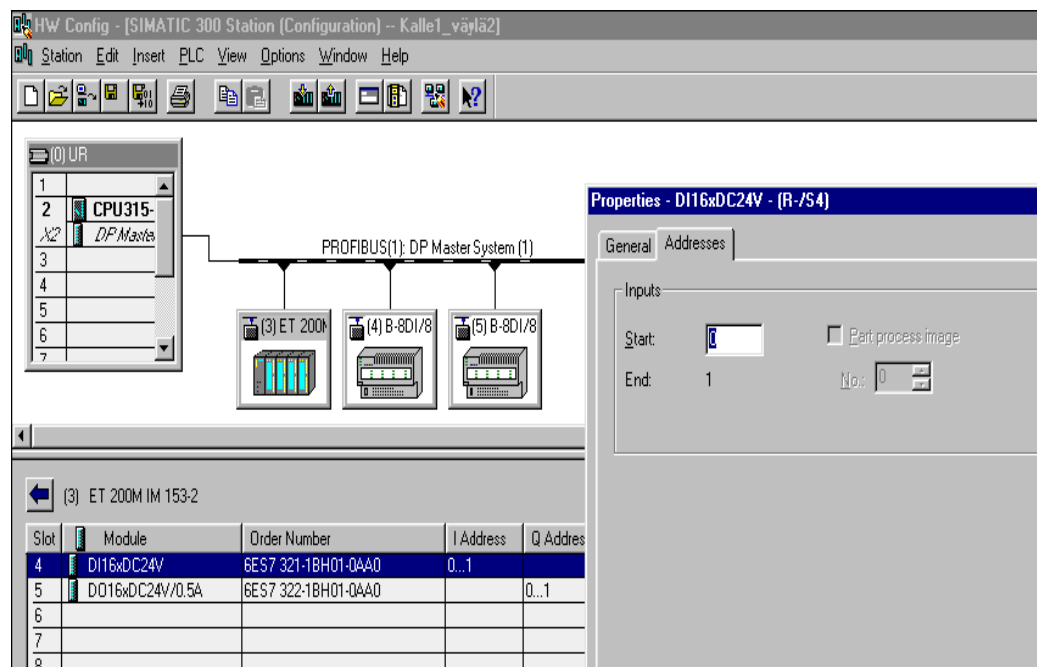


Kuva 21 Konfigurointi /1/

2.14 Väylämoduulien osoitteiden asettaminen

Viimeisenä varsinaisena konfigurointitehtävänä suoritetaan tavuosoitteiden asettaminen I/O moduuleihin. Klikataan sen I/O -moduulin rivivalikko aktiiviseksi, se parametroidaan, jolloin avautuu kyseisen moduulin *Properties*-ikkuna (kuva 22). Tässä esimerkissä asetetaan ET 200M yksikön Input osoitteet. ET 200 M Input -moduulissa on 16 sisäänmenoa eli ne muodostavat 2 tavua. Ensimmäiseksi tavuosoitteeksi annetaan 0, jolloin seuraava

muodostuu automaattisesti 1:ksi. Yksinkertaisuuden vuoksi kannattaa antaa ET 200 M Output osoitteiksi samat numerot. /1/



Kuva 22 Moduulin Properties -ikkuna

ET 200 B moduuleille pitää antaa myös tavuosoitteet, ja ne voidaan antaa esim. seuraavan listan mukaisesti:

1. ET 200 M (väyläosoite 3)
2. input 0 ja 1
3. output 0 ja 1
4. ET 200 B (väyläosoite 4)
5. input 2
6. output 2. /1/

Käytettävissä olevat bittiosoitteet ovat seuraavan listan mukaiset:

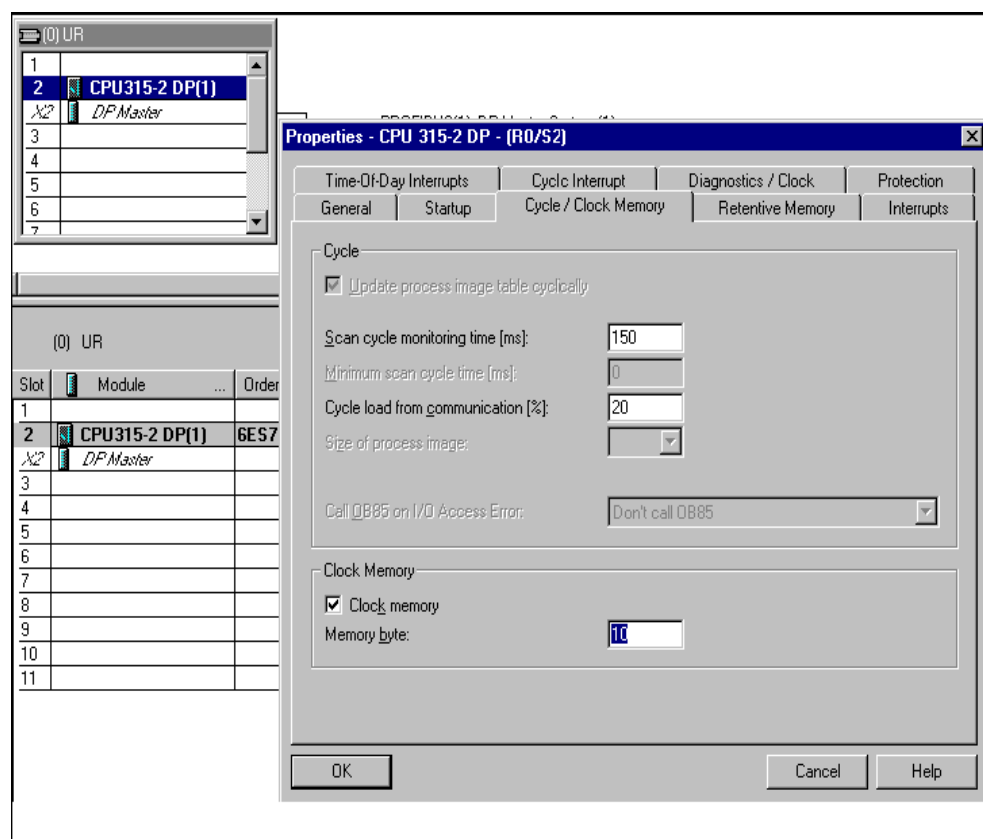
1. ET 200 M (väyläosoite 3)
2. E 0.0 – E 0.7
3. E 1.0 – E 1.7
4. A 0.0 – A 0.7
5. A 1.0 – A 1.7

6. ET 200 B (väyläosoite 4)
7. E 2.0 – E 2.7
8. A 2.0- A 2.7 /1/

Tarvitset näitä bittiosoitteita tehdessäsi ohjelmaa logiikalle.

2.15 Clock Memoryn asettaminen

Kaksoisklikataan UR-ikkunassa CPU 351 riviä. Näin aukeaa properties-ikkuna, josta valitset cycle/clock memory välilehden. Klikataan clock memory -ruutuun rasti. Asetetaan memory byte -arvoksi 10. Katso kuva 23. /1/

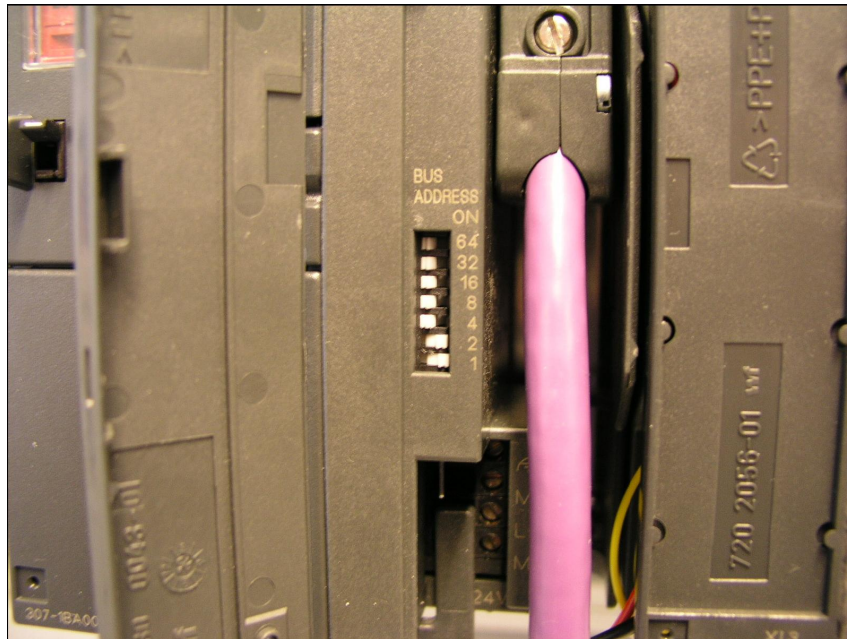


Kuva 23 Clock Memoryn asettaminen /1/

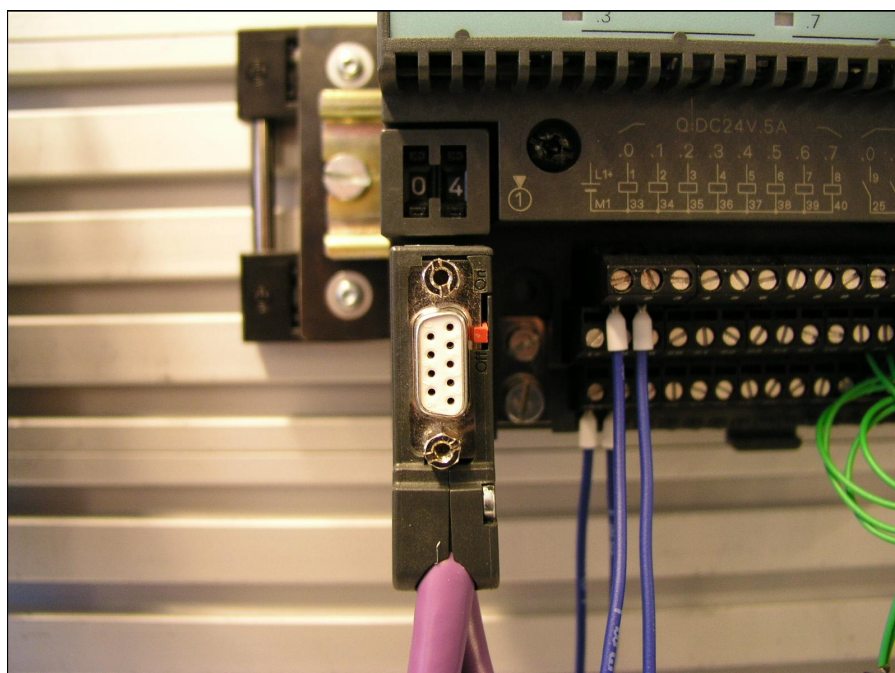
MUISTA TALLENTAA!

2.16 Konfiguroinnin lataaminen CPU:n muistiin

Latauksen onnistumiseksi pitää suorittaa kaikki kaapeloinnit valmiiksi järjestelmässä, jotta vältetään turhilta virhetilanteilta. Seuraavat kaapelointien pitää olla suoritettuna. Asetetaan kytkimet I/O -moduulien väyläosoitteita vastaaviksi. (kuvat 24 ja 25) /1/



Kuva 24 Kytkimen asetukset



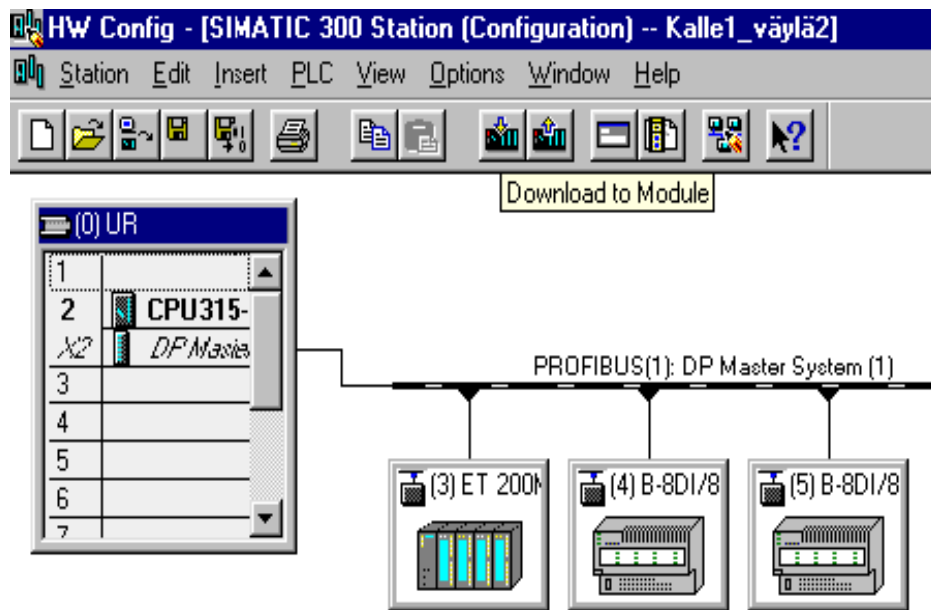
Kuva 25 Kytkimen asetus

Ennen konfiguroinnin latausta logiikka on resetoitava. Käännetään ohjelmakytin MRES-asentoon ja pidetään se siellä niin kauan, että STOP-valo vilkkuu nopeasti. Jos valo ei vilku nopeasti, niin vapautetaan kytkin ja toista toimenpide. Katso kuva 26.

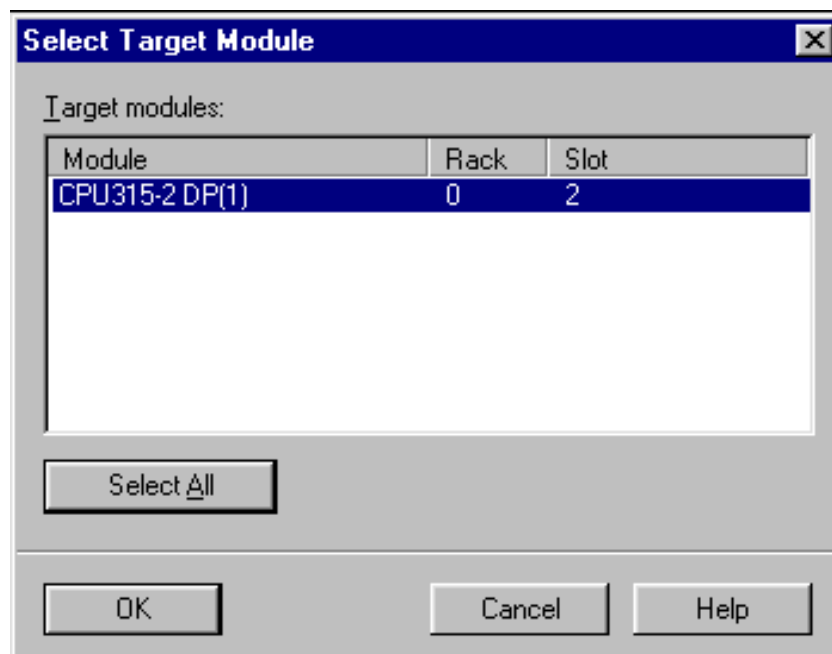


Kuva 26 Logiikan resetointi

Konfiguroinnin lataus (Download) suoritetaan työkalurivin ”*Download to Module*” painikkeella (Kuva 27). Näyttöön ilmestyy Select Target Module -ikkuna (kuva 28), josta valitaan *OK* -painike, kuten myös seuraaviin kysymyksiin, joita tulee latauksen yhteydessä./1/



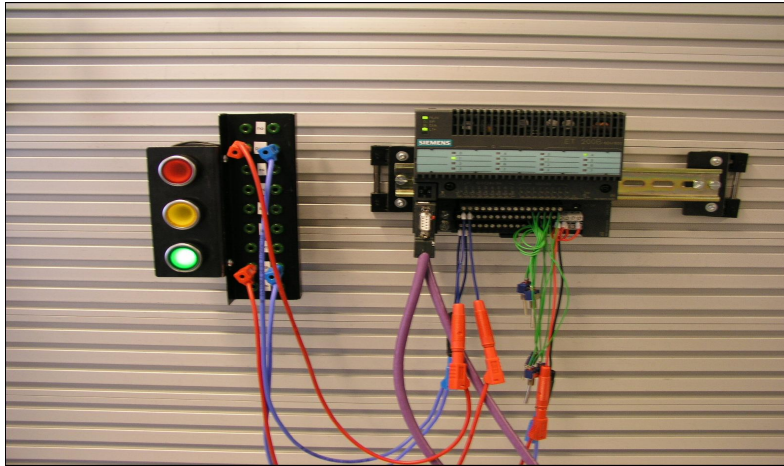
Kuva 27 Konfiguroinnin lataus /1/



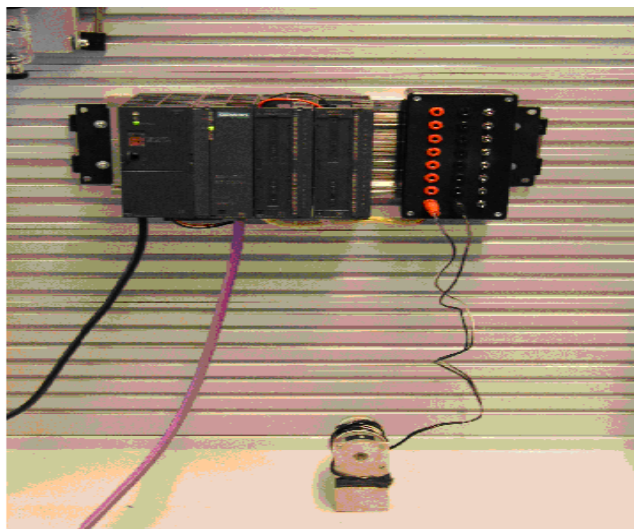
Kuva 28 Avautuvista ikkunoista painetaan OK-painiketta /1/

Seuraavaksi tarvitaan DC-moottorin ja ”liikennevalot”. Katso kuvat 29 ja 30. Tehtävänäön laatia blockiin OB1 ohjelma, joka pyörittää moottoria, kun kummallakin I/O moduulilla on yksi tulo asetettu aktiiviseksi. Tuloiksi saadaan valita haluamamme. Tulojen ja lähtöjen osoitteet löytyvät sivulta 18. Kun moottori pyörii, niin vihreä valo palaa. Kun moottori ei pyöri, punainen valo

vilkkuu. Johdotetaan valot haluamiisi lähtöihin ET 200 B:ssä. Punaisen valon vilkkuminen saadaan aikaan käyttämällä edellä määritettyä merkkiä M 10.0 – M 10.7. Osoitteen muuttaminen tällä välillä vaikuttaa vilkkumisen taajuuteen. Moottori johdotetaan ET 200 M:n lähtöön. /1/



Kuva 29 ”Liikennevalot”



Kuva 30 DC-moottori

3 ASI-VÄYLÄN OHJE

3.1 Väylämoduulien alustus

Aloitetaan väylän rakentaminen asettamalla väylämoduuleille väyläosoitteet. Osoitteen asettaminen tapahtuu ohjelmointilaitteella. Osoitteeksi voidaan asettaa mikä numero tahansa. Kuitenkaan väylässä ei saa olla kahta samaa osoitetta. Valmiiksi ohjelmoituista moduuleista löydetään osoitteet komponentin kyljessä olevista tarroista. Näiden moduulien osoitteita ei tarvitse muuttaa. /3/



Kuva 30 Moduuli on kytketty ohjelmointilaitteeseen /3/

Aloitetaan osoitteen ohjelmoiminen kytkemällä jännite ohjelmointilaitteelle. Kytketään moduuli ohjelmointilaitteeseen (kuva 30). Osoitetta voi muuttaa nuolinäppäimillä (Kuva 31). Haluttu osoite saadaan siirrettyä komponentin muistiin painamalla Write -näppäintä. Read-näppäimellä voi lukea, mikä osoite komponenttiin on ohjelmoitu. Komponentille on annettu väyläosoitteeksi 16 (kuva 31). /3/



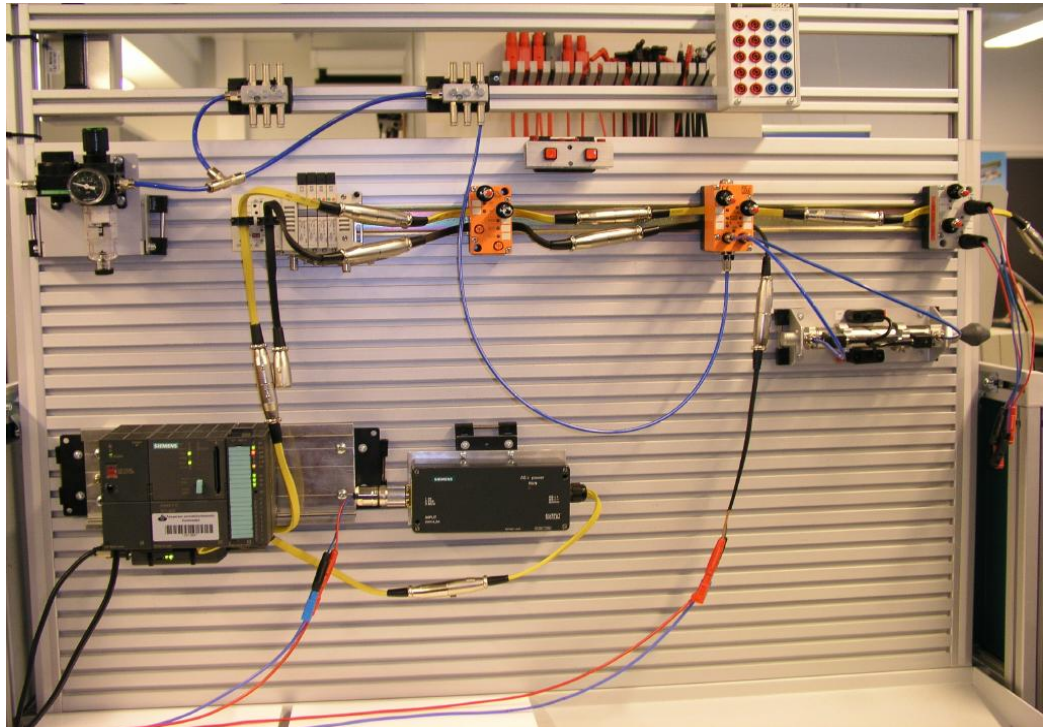
Kuva 31 komponentin väyläosoitteeksi on annettu 16 /3/

3.2 Järjestelmän pystyttäminen

Asi-väylään tarvittavat komponentit löydetään kaapista PLC 2.
Seuraavat laitteet tarvitaan järjestelmään:

1. logiikka Siemens S7 300 (hyllypaikka PLC 2-4)
2. Asi power (hyllypaikka Asi 1 tai 2) ja syöttöjohdot
3. relelähdot (hyllypaikka Asi 3)
4. tarvittava määrä Asi-moduleita (hyllypaikka Asi-modulit)
5. PC, johon on asennettu Simatic Manager ohjelmisto
6. PC:n ja logiikan välinen kaapeli
7. paineilmasylinteri
8. väyläkaapeli. /3/

Katso kuvat 32-35



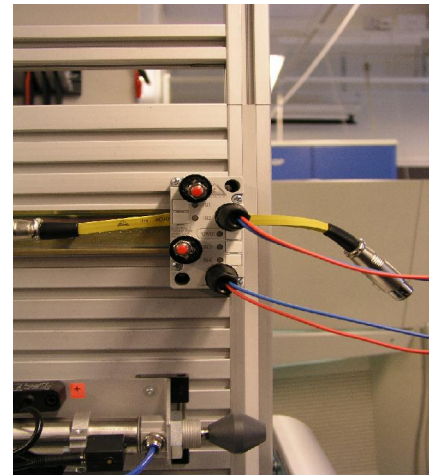
Kuva 32 Asi-väylään tarvittavat komponentit /3/



Kuva 33 Paineilmasyylinteri /3/



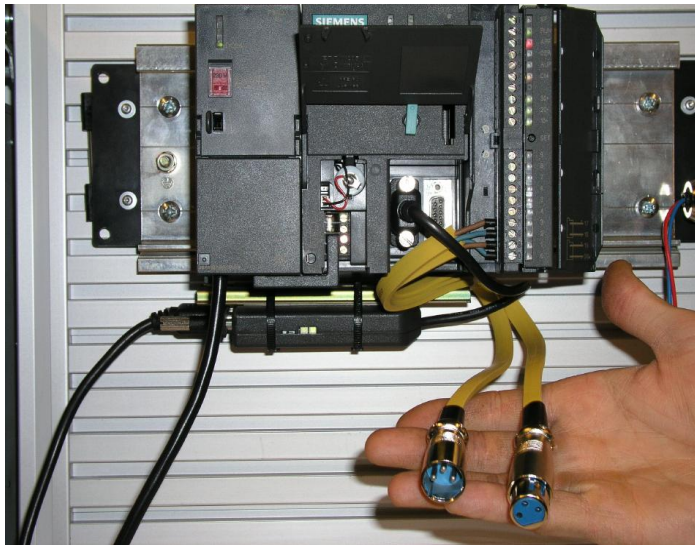
Kuva 34 Asi-moduuli



Kuva 35 /3/

3.3 Asennus

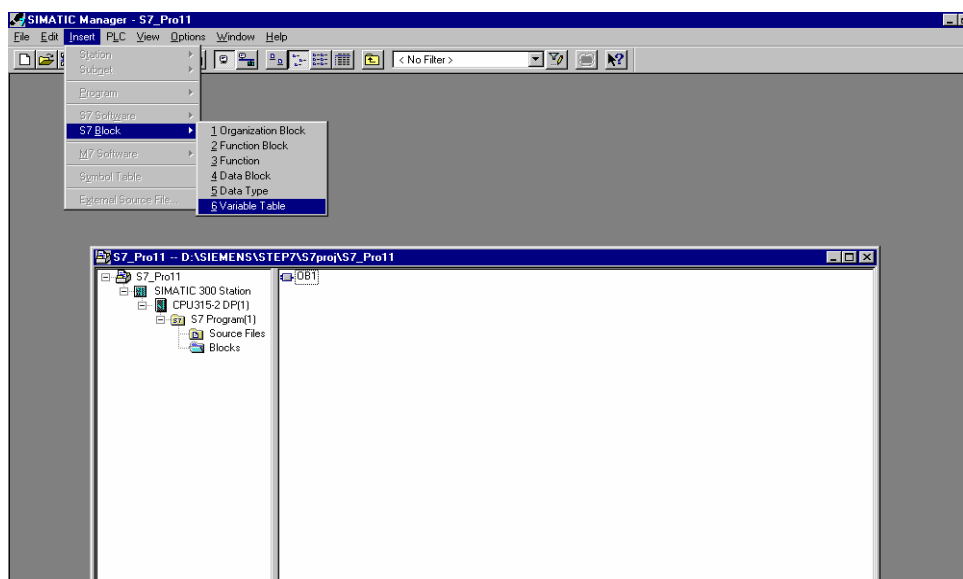
Kiinnitetään tarvittavat komponentit Boschin asennuspöytiin. Kytetään syöttöjännite logiikalle. Kytetään syöttöjännite Asi-Powerille 24 V. Huomaa, että Asi-Power tarvitsee toimiakseen vähintään 4.2 ampeerin virran. Kytetään väyläkaapelit. Keltainen kaapeli liitetään keltaisiin liittimiin ja musta mustiin. Komponenttien järjestyksellä ei ole merkitystä. Kytetään logiikan ja PC:n välinen adapteri siten, että MPI puoli (ilmoitettu adapterin kannessa) kiinnitetään logiikkaan ja RS232- puoli PC:n sarjaliittimeen. (Kuva 36) Valitaan adapterin nopeudeksi 19.2 kb/s (kytkin adapterin sivussa). Käynnistetään PC ja avataan Simatic Manager –ohjelma. /3/



Kuva 36 Simatic S7 300

3.4 Projektin luominen

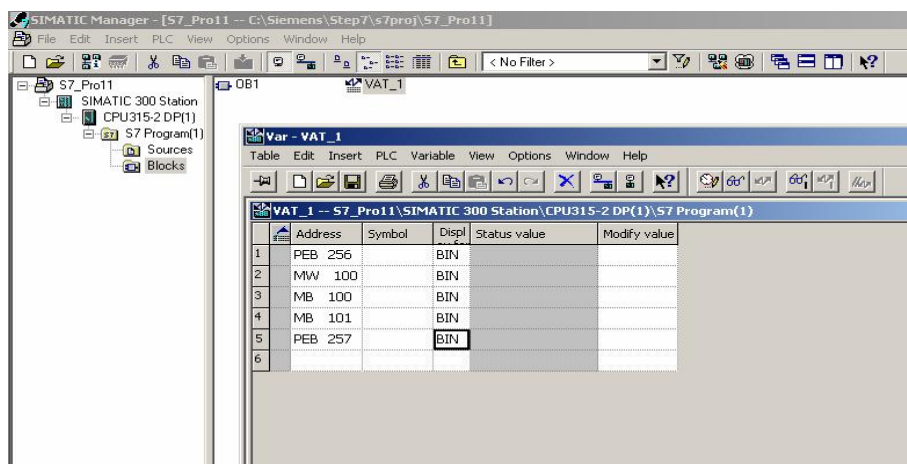
Ensimmäiseksi luodaan uusi projekti Wizardin ohjeiden mukaan ja nimetään se haluamallamme nimellä. Valitaan logiikaksi CPU315-2DP. Seuraavaksi luodaan Variable tables (VAT). Valitaan Insert -alasvetovalikosta S7 Block ja sen alta Variable tables (Kuva 37). OB1 -kuvake ei saa olla ruudussa valittuna eli aktiivisena. Hyväksytään Variable tables klikkaamalla OK:ta. /3/



Kuva 37 Variable tablen luominen /3/

Ruutuun ilmestyy OB1 -kuvakkeen viereen VAT 1 -kuvake. Klikkaamalla tätä aukeaa kuvan 38 näkymä. Kirjoitetaan Address -sarakeeseen PEB 256. Muutetaan Monitor Format sarakeeseen BIN klikkaamalla HEX -ruutua. Ladataan ohjelma logiikalle Download -painikkeella. Laitamme Monitor tilan päälle painamalla silmälasikuvaketta. Nyt nähdään, mikä bitti muuttuu, kun painamme moduulin Asi os. 1 in -painiketta. Moduulin osoitteeksi on määritetty 1 ohjelmointilaitteella. Osoitteet on luettavissa tarroista moduuleiden kyljistä.

/3/



	Address	Symbol	Displ	Status value	Modify value
1	PEB 256		BIN		
2	MW 100		BIN		
3	MB 100		BIN		
4	MB 101		BIN		
5	PEB 257		BIN		
6					

Kuva 38 Variable Table

3.5 Osoitteet

ASI-väylässä yksi moduuli käyttää neljää bittiä. 16 bittiä muodostaa sanan. Simatic ohjelmassa tulospaikka merkitään lyhenteellä PEW. Lähtöpaikka merkitään lyhenteellä PAW. MB eli merkkiritavu näyttää 8 bittiä. Taulukosta 1 voidaan lukea moduulien osoitteet. Korttipaikassa 1 N kirjain vastaa osoitetta 256. Jos moduulin kylkeen on kirjoitettu osoite 1, löytyvät sen neljä bittiä tavusta 257 (256+1) (taulukko 1). Asi os.1 IN 1 = slave 3 bitti 0, IN 2 = slave 3 bitti 1 (kuva 39). Jos Asi os. 21, niin löytyvät sen neljä bittiä tavusta 266 (256+10) (taulukko 1). Ja sen IN 1 = slave 21 bitti 0 (kuva 40). /3/

Taulukko 1 Moduulien osoitetaulukko /3/

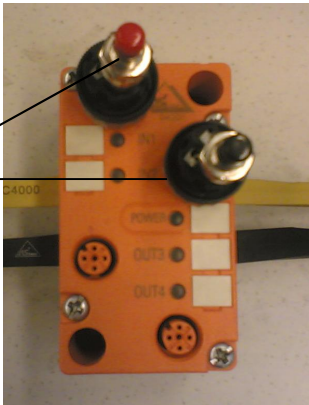
Assignment:

I/O byte number	Bit 7-4	Bit 3-0
	reserved	Slave 1 Bit 3 Bit 2 Bit 1 Bit 0
n+0	reserved	Slave 1 Bit 3 Bit 2 Bit 1 Bit 0
n+1	Slave 2	Slave 3
n+2	Slave 4	Slave 5
n+3	Slave 6	Slave 7
n+4	Slave 8	Slave 9
n+5	Slave 10	Slave 11
n+6	Slave 12	Slave 13
n+7	Slave 14	Slave 15
n+8	Slave 16	Slave 17
n+9	Slave 18	Slave 19
n+10	Slave 20	Slave 21
n+11	Slave 22	Slave 23
n+12	Slave 24	Slave 25
n+13	Slave 26	Slave 27
n+14	Slave 28	Slave 29
n+15	Slave 30 Bit 3 Bit 2 Bit 1 Bit 0	Slave 31 Bit 3 Bit 2 Bit 1 Bit 0

n = Start address

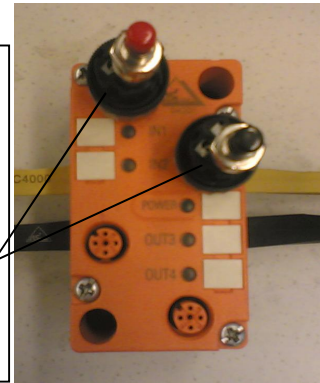
Lähtötavu PEB 257 pitää sisällään Slave 2:n ja 3:n. Asi master käsittelee kokonaista sanaa, jolle tiedot tulevat ”slaveilta” neljän bitin ryhminä. Katso taulukko 1 ja kuva 39. /3/

PEB 256 (N)	7	6	5	4	3	2	1	0
PEB 257 (N+1)	7	6	5	4	3	2	1	0
JNE...(N+X)	7	6	5	4	3	2	1	0
PEB 266 (N+10)	7	6	5	4	3	2	1	0



Kuva 39 Asi os.1 IN 1 = slave2 bitti 0

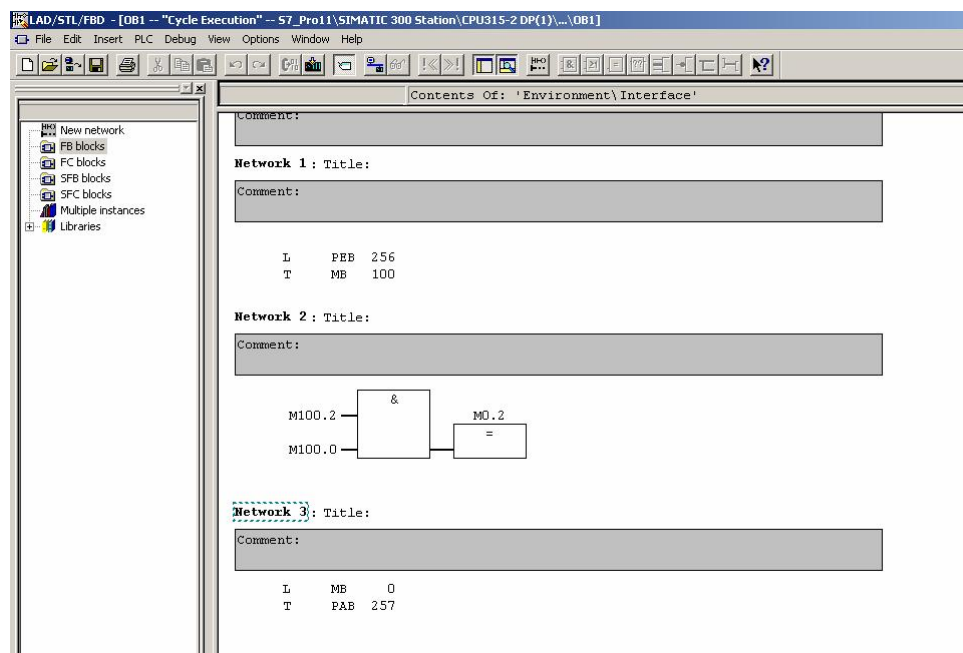
PEB 256 (N)	7	6	5	4	3	2	1	0
PEB 257 (N+1)	7	6	5	4	3	2	1	0
JNE...(N+X)	7	6	5	4	3	2	1	0
PEB 266 (N+10)	7	6	5	4	3	2	1	0



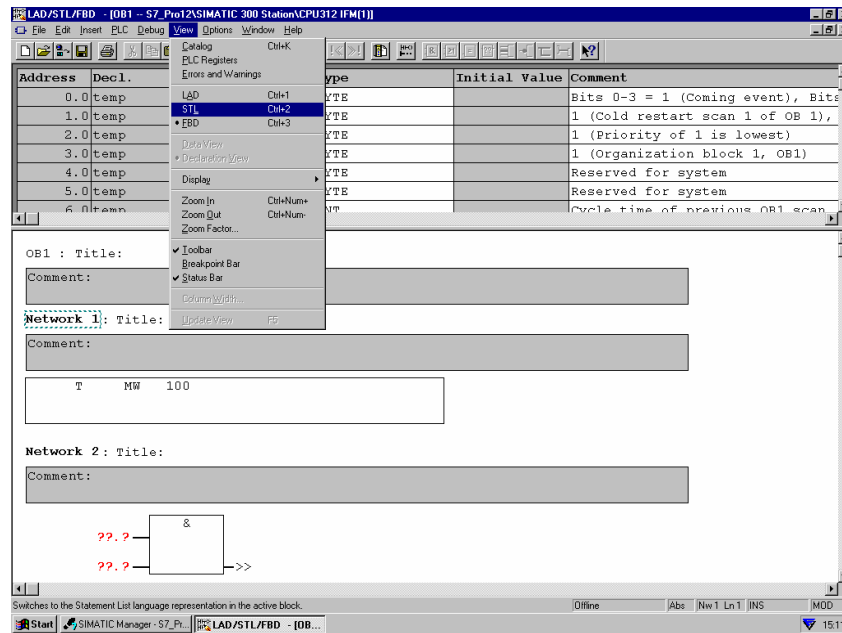
Kuva 40 Asi os. 21 IN 1 = slave 21 bitti 0

3.6 Ohjelmointi

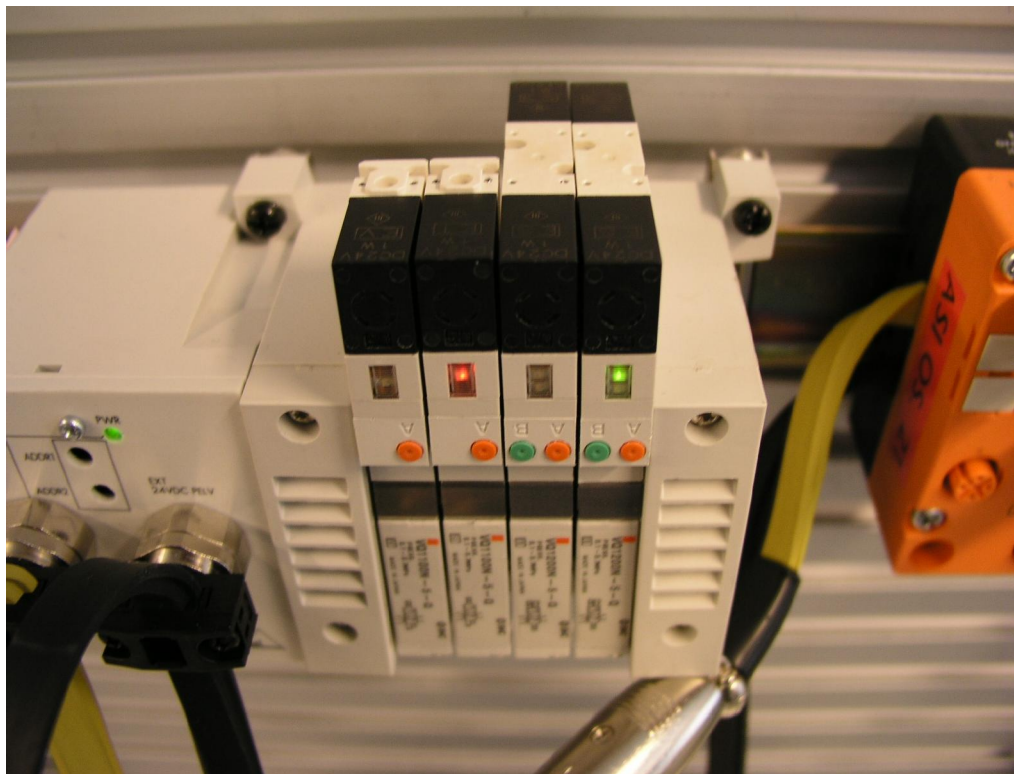
Kuvassa 41 on ohjelma, joka laittaa lähdön 2 päälle moduulilla Asi os. 02-03 (Kuva 43), kun moduulin Asi os. 01 tuloihin IN 1 ja IN 3 vaikutetaan (piiri 2). Piirissä 1 ladataan (L) sana PEW 256 logiikan akkuun. Tämän jälkeen PEB 256 siirretään (T) merkkerisanaan 100. Vasta sen jälkeen sanan 256 tietoja voidaan käyttää ohjelmassa, merkkerin avulla. Piirissä 3 ladataan merkkerisana 0 logiikan akkuun ja siirretään lähtösana PAW 257. Pystyäksemme tekemään lataus- ja siirtokäskyt meidän täytyy valita View -alasvetovalikosta STL - ohjelmointitapa. Logiikkakaavioita pystytään tekemään FBD - ohjelmointitavalla, katso kuva 42. /3/



Kuva 41 Logiikkakaavio /3/



Kuva 42 Logiikkakaavion tekeminen /3/



Kuva 43 Moduuli Asi os. 02-03

4 TOIMINTAYKSIKÖT (Function Block, FB)

4.1 Ohjelmarakenne

Useimmilla automaatiojärjestelmillä on mahdollisuus tehdä ohjelma lineaarisesti tai jaotellusti. Lineaarisella ohjelmalla on yksinkertainen rakenne. Kaikki ohjelman käskyt ovat samassa ohjelmatiedostossa ja ne suoritetaan perä jälkeen rivi riviltä. Tämä on lähes aina OB1. /2/

Kaikissa ohjelmissa on oltava organisaatioyksikkö (OB). OB:t edustavat korkeinta ohjelmatasoa. Yksinkertaiset ohjelmat joissa halutaan käyttää lineaarista rakennetta, voidaan kirjoittaa kokonaisuudessaan organisaatioyksikköön. Yleensä kuitenkin organisaatioyksikköön kirjoitetaan ehdollisia tai ehdottomia kutsuja muihin yksiköihin (kuva 44) (FB:t, FC:t, OB:t), eli käytetään jaoteltua ohjelmointimallia. /2/

4.2 Toimintayksiköt (FB)

Toimintayksiköt (FB) ovat alemman tason yksiköitä. Niihin kirjoitetaan ohjelman osat, joita voidaan kutsua useita kertoja ohjelmakierron aikana. Näin säästytään kirjoittamasta ohjelmaa uudelleen ja uudelleen. Jokaiselle toimintayksikölle on osoitettava oma tiedostoyksikkö (DB), jonne kaikki sen tarvitsemat muuttujat ja data on tallennettuna. Funktiot (FC) eivät tarvitse omaa tiedostoyksikköä. /2/

SIEMENS

Yksikkölajit ja -toiminnot

Yksikkölaji	Ominaisuudet
Organisaatio-yksikkö (OB)	<ul style="list-style-type: none"> - Käyttäjän liitoskohta käyttöjärjestelmään - Porrastettu prioriteetti (0...25) - Erityisinformatiota paikallisrekisterissä
toiminta-yksikkö (FB)	<ul style="list-style-type: none"> - Parametroitavissa / muistilla - Ei parametroitavissa / muistilla - Ei parametroitavissa / ilman muistia
(Ohjaus-) Toiminta (FC)	<ul style="list-style-type: none"> - Takaisinantoarvo (Parametrit tulee kytkeä kutsuttaessa) - Periaatteessa ilman muistia - Parametroitavissa
Tiedosto-yksikkö (DB)	<ul style="list-style-type: none"> - Rakenteinen paikallinen datatalletus (Oheis- DB) - Rakenteinen globaali datatalletus (käytettävissä koko ohjelmassa)

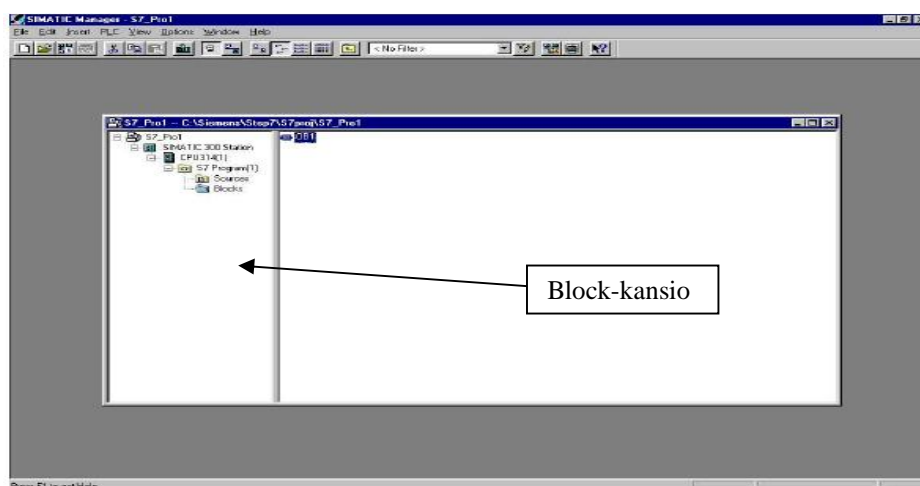
Kuva 44 Yksikkölajit ja -toiminnot /2/

4.3 Toimintayksikön luonti

Katsotaan SIMATIC STEP 7 Ohjelmointiohje→ Projektin muodostus.

Blocks – kansioon(kuva 45) tallennetaan kaikki luodut yksiköt (OB, FB, FC).

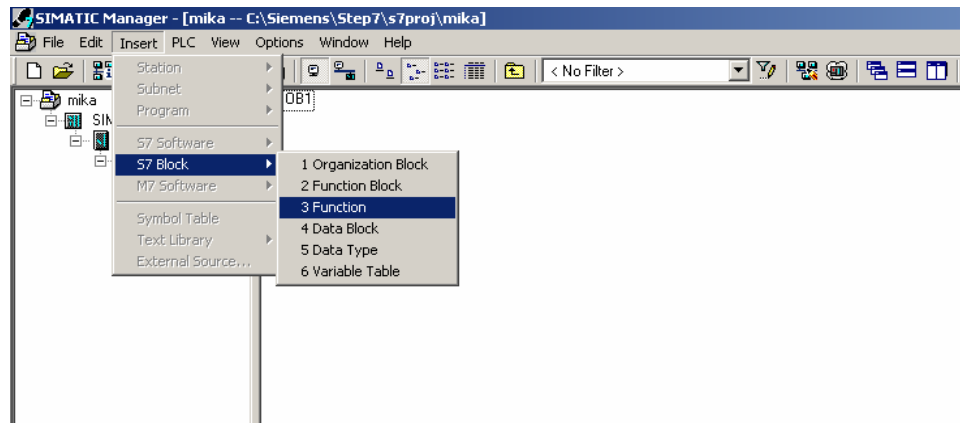
/2/



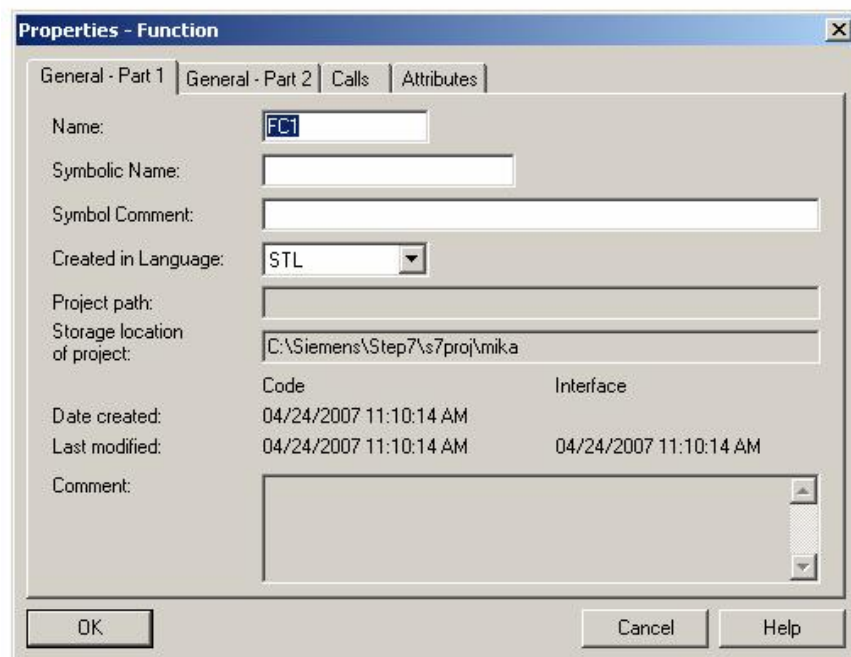
Kuva 45 Toimintayksikön luonti

4.4 S7-yksikön lisäys

Seuraavaksi valitaan insert- alaspöytävalikosta Function (kuva 46). Tämän jälkeen avautuu Properties-Function ikkuna (kuva 47). Tässä ikkunassa voidaan nimetä FC. /2/

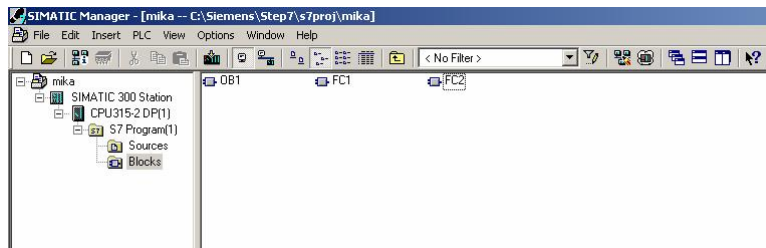


Kuva 46 Function -valinta



Kuva 47 FC:n nimeäminen

Luodut FC:t tulevat näkyviin OB1:n viereen (kuva 48). FC:t avautuvat klikkaamalla niitä. /2/

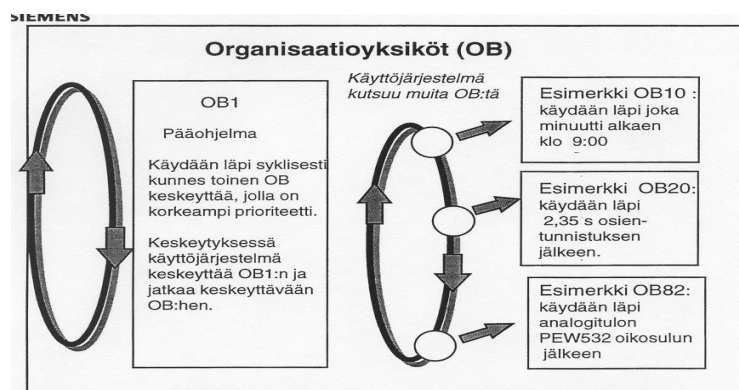


Kuva 48 FC:t syntyvät OB1:n viereen

5 KESKEYTYKSET

Organisaatioyksiköt (Organisation Block, OB) muodostavat liitäntäpisteen S7 CPU:n (keskussyksikön) ja ohjelman välillä. Käyttöjärjestelmä kutsuu koodiyksikköä. OB1 on syklinen yksikkö ja se läpikäydään joka sykli. OB1:een voidaan tallentaa koko ohjelma, tai ohjelma voidaan jakaa yksittäisiin logiikkayksiköihin ja OB1:tä käytetään vain näiden yksiköiden kutsumiseksi (kuva 49). OB1:n lisäksi käyttöjärjestelmä voi kutsua myös muita OB:itä tietyn toiminnan saavuttamiseksi. OB:tä voi kutsua vain käyttöjärjestelmä, ei mikään muu yksikkö. S7 CPU:n käyttöjärjestelmässä on 26 prioriteettiluokkaa, jolloin jokainen prioriteettiluokka käyttää OB:tä liitäntäpisteenä. Käyttöjärjestelmä suorittaa OB:t prioriteettiluokkansa mukaan. OB:llä (Prioriteettiluokka 1) on alhaisin prioriteetti kaikista OB:stä, mikä merkitsee, että kaikki muut OB:t saavat keskeyttää OB1:n tarvittaessa (Kuva 49). Jokaisella S7 CPU:lla on erilaisia OB:ita ja vastaavat toiminnot. Eri OB:illa on seuraavat erityistoiminnot:

1. OB1 Sykli, Jatkuva silmukka
2. OB10 Kelloaika, Alkaa tietyllä hetkellä
3. OB20 Viivekeskeytys, Suoritus ohjelmoidun ajan jälkeen
4. OB35 Aikakeskeytys, Toisto tietyin aikavälein
5. OB40 Prosessikeskeytys, Suoritus keskeytyksessä
6. OB80 – 87 Virhe OB (Asynk.virhe), Suoritus Asynkroisvirheestä
7. OB121,122 , Virhe OB (Synk.virhe), Suoritus synkroisvirheestä
8. OB101 Jälleenkäynnistys, Suoritus uudelleen käynnistäessä. /2/



Kuva 49 OB1 kutsuu yksittäisiä logiikkayksiköitä /2/

6 DATATYYPIN MUUNNOS

S7-300 ja S7-400 -laitteiden käskykannassa on monipuoliset muunnosmahdollisuudet. Jokaisella käskyllä on seuraava muoto.

IN= tuloarvo

OUT= lähtöarvo /2/

6.1 Datatyyppin muunnos operaatiot

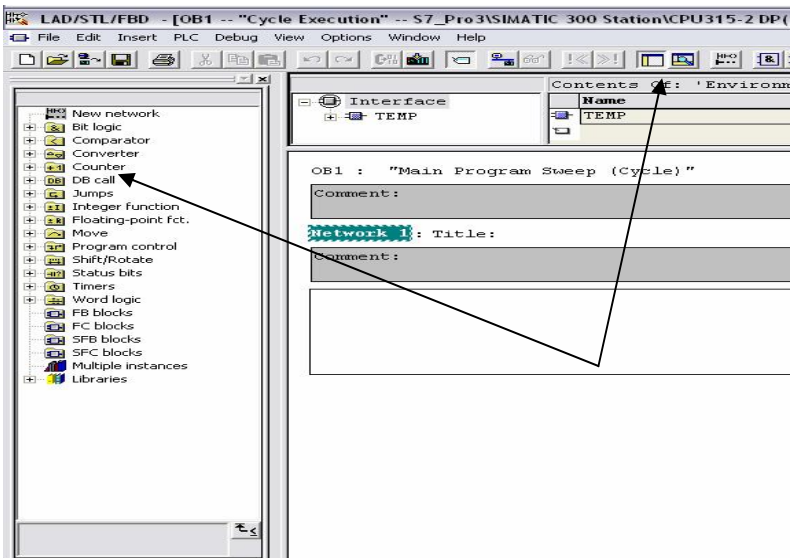
Seuraavilla operaatioilla voidaan binäärikoodatut desimaali- ja kokonaisluvut muuntaa muihin lukumuotoihin:

1. BCD_I BCD-luvun muunto kokonaisluvuksi (16 bittiä)
2. I_BCD kokonaisluvun (16 bittiä) muunto BCD-luvuksi
3. BCD_DI BCD-luvun muunto kokonaisluvuksi (32 bittiä)
4. I_DI kokonaisluvun (16 bittiä) muunto kokonaisluvuksi (32 bittiä)
5. DI_BCD kokonaisluvun (32 bittiä) muunto BCD-luvuksi
6. DI_R kokonaisluvun (32 bittiä) muunto liukuluvuksi. /2/

6.2 Operaation valinat Simatic Manager -ohjelmassa


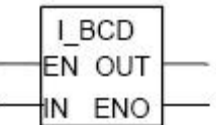
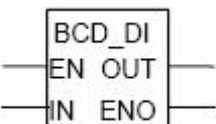
Käynnistetään Simatic Manager –ohjelma Wizard -ohjeiden mukaisesti.

Käskyt (Taulukot 2-5) löytyvät Simatic- ohjelmasta kohdasta Program Elements
→ Converter (kuva 50).

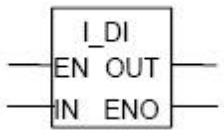
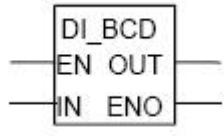
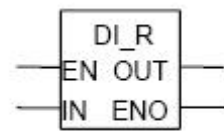


Kuva 50 Converter -käskyn valinta

Taulukko 2 Datatyyppin muunnos käskyjen kuvaukset /2/

Käsky	Kuvaus
	Operaatio BCD-luvun muunto kokonaisluvuksi (16 bittiä) lukee tuloparametrin IN sisällön kolminumeroisena binäärikoodattuna desimaalilukuna (BCD, + 999) ja muuntaa tämän luvun kokonaislukuarvoksi (16 bittiä). Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.
	Operaatio kokonaisluvun (16 bittiä) muunto BCD-luvuksi lukee tuloparametrin IN sisällön kokonaislukuarvona (16 bittiä) ja muuntaa sen kolminumeroiseksi binäärikoodatuksi desimaaliluvuksi (BCD, + 999). Tulosta voi kysyä lähdöstä OUT. Jos ilmenee ylivuoto, ENO = 0.
	Operaatio BCD-luvun muunto kokonaisluvuksi (32 bittiä) lukee tuloparametrin IN sisällön seitsennumeroisena binäärikoodattuna desimaalilukuna (BCD, + 9 999 999) ja muuntaa tämän luvun kokonaislukuarvoksi (32 bittiä). Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.

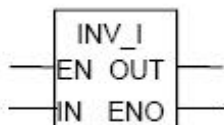
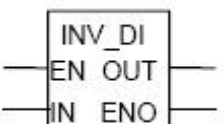

Taulukko 3 Datatyypinmuunnos operaatioiden kuvaus /2/

Käsky	Kuvaus
	<p>Operaatio kokonaisluvun (16 bittiä) muunto kokonaisluvuksi (32 bittiä) lukee tuloparametrin IN sisällön kokonaislukuna (16 bittiä) ja muuntaa tämän kokonaisluvuksi (32 bittiä). Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.</p>
	<p>Operaatio kokonaisluvun (32 bittiä) muunto BCD-luvuksi lukee tuloparametrin IN sisällön kokonaislukuarvona (32 bittiä) ja muuntaa tämän arvon seitsennumeroiseksi binäärikoodatuksi desimaaliluvuksi (BCD, + 9 999 999). Tulosta voi kysyä lähdöstä OUT. Jos ilmenee ylivuoto, ENO = 0.</p>
	<p>Operaatio kokonaisluvun (32 bittiä) muunto liukuluvuksi lukee tuloparametrin IN sisällön kokonaislukuna (32 bittiä) ja muuntaa tämän liukuluvuksi. Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.</p>

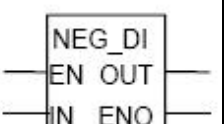
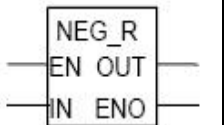
Seuraavilla operaatioilla voidaan muodostaa kokonaislukujen komplementteja tai vaihtaa liukuluvun etumerkkiä:

1. INV_I Kokonaisluvun (16 bittiä) yhden komplementin muodostus
2. INV_DI Kokonaisluvun (32 bittiä) yhden komplementin muodostus
3. NEG_I Kokonaisluvun (16 bittiä) kahden komplementin muodostus
4. NEG_DI Kokonaisluvun (32 bittiä) kahden komplementin muodostus
5. NEG_R Liukuluvun etumerkin vaihto. /2/

Taulukko 4 Datatyyppin muunnoskäskyjen kuvaus /2/

Käsky	Kuvaus
	<p>Operaatio kokonaisluvun (16 bittiä) yhden-komplementin muodostus lukee tuloparametrin IN sisällön ja suorittaa boolean sanalukitusoperaation 16 bitin EHDOTON TAI lukitus heksadesimaalikaaviolla FFFFH. Täten jokaisen bitin arvosta tulee käänteinen. Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.</p>
	<p>Operaatio kokonaisluvun (32 bittiä) yhden-komplementin muodostus lukee tuloparametrin IN sisällön ja suorittaa boolean sanalukitusoperaation EHDOTON TAI lukitus heksadesimaalikaaviolla FFFF FFFFH. Täten jokaisen bitin arvosta tulee käänteinen. Tulosta voi kysyä lähdöstä OUT. ENO:n signaalitila on aina sama kuin EN:n.</p>
	<p>Operaatio kokonaisluvun (16 bittiä) kahden-komplementin muodostus lukee tuloparametrin IN sisällön ja kääntää etumerkin ympäri (esim. positiivisesta arvosta negatiiviseksi arvoksi). Tulosta voi kysyä lähdöstä OUT. EN:n ja ENO:n signaalitila on aina sama, seuraavalla poikkeuksella: Mikäli EN:n signaalitila on "1" ja esiintyy ylivuoto, on ENO:n signaalitila "0"</p>

Taulukko 5 Datatyyppin muunnoskäskyjen kuvaus /2/



Käsky	Kuvaus
	<p>Operaatio kokonaisluvun (32 bittiä) kahden-komplementin muodostus lukee tuloparametrin IN sisällön ja kääntää etumerkin ympäri (esim. positiivisesta arvosta negatiiviseksi arvoksi). Tulosta voi kysyä lähdöstä OUT. EN:n ja ENO:n signaalitila on aina sama, seuraavalla poikkeuksella: Mikäli EN:n signaalitila on "1" ja esiintyy ylivuoto, on ENO:n signaalitila "0"</p>
	<p>Operaatio liukuluvun etumerkin vaihto lukee tuloparametrin IN sisällön ja kääntää etumerkkibitin ympäri, ts. operaatio muuttaa luvun etumerkin (esim. positiivisesta 0:sta negatiiviseksi 1:ksi). Eksponentin ja mantissan bitit pysyvät muuttumattomina. Tulosta voi kysyä lähdöstä OUT. EN:n ja ENO:n signaalitila on aina sama, poikkeus: Mikäli EN:n signaalitila on "1" ja esiintyy ylivuoto, on ENO:n signaalitila "0".</p>

Seuraavilla operaatioilla voidaan liukuluku (32 bittiä, IEEE-FP) muuntaa kokonaisluvuksi (32 bittiä) (taulukot 6 ja 7). Yksittäiset operaatiot eroavat pyöristyslajiltaan toisistaan. /2/

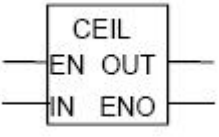
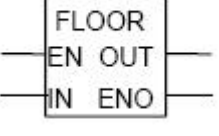
1. ROUND Luvun pyöristys
2. TRUNC Kokonaisluvun muodostus
3. CEIL Seuraavaksi suuremman kokonaisluvun muodostus liukuluvusta
4. FLOOR Seuraavaksi pienemmän kokonaisluvun muodostus liukuluvusta.

/2/

Taulukko 6 Datatyypin muunnoskäskyjen kuvaus /2/

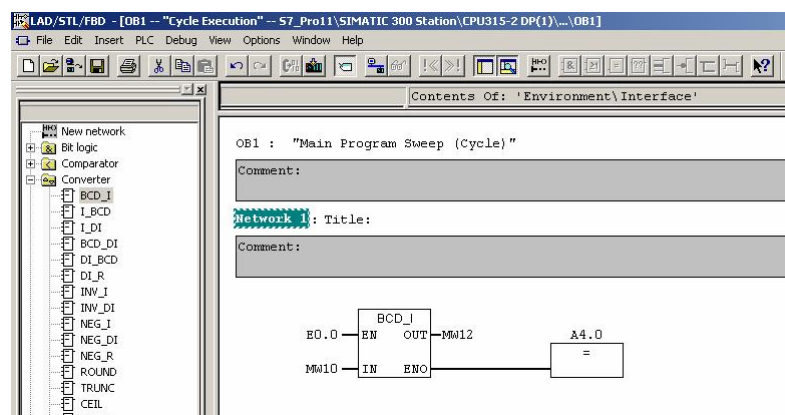
Käsky	Kuvaus
	<p>Operaatio luvun pyöristys lukee tuloparametrin IN sisällön liukulukuna ja muuttaa sen kokonaisluvuksi (32 bittiä). Tulos on lähin lähtöparametrin OUT ilmoittama kokonaisluku. Mikäli murto-osa = x,5, niin parillinen luku pyöristetään alaspäin (esimerkki: 2,5 -> 2, 1,5 -> 2). Mikäli esiintyy ylivuoto, ENO = 0. Mikäli tulo ei ole liukuluku, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
	<p>Operaatio kokonaisluvun muodostus lukee tuloparametrin IN sisällön liukulukuna ja muuntaa tämän kokonaisluvuksi (32 bittiä) (esimerkki: 1,5:stä tulee 1). Tulos on lähdessä OUT ilmoitettava liukuluvun kokonaisosa. Mikäli esiintyy ylivuoto, ENO = 0. Mikäli tulo ei ole liukuluku, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>

Taulukko 7 Datatyyppin muunnos käskyjen kuvaus /2/

Käsky	Kuvaus
	<p>Operaatio seuraavaksi suuremman kokonaisluvun muodostus liukuluvusta lukee tuloparametrin IN sisällön liukulukuna ja muuntaa tämän kokonaisluvuksi (32 bittiä) (esimerkki: +1,2 -> +2; -1,5 -> -1). Tulos on pienin kokonaisluku, joka on annettua liukulukua suurempi/yhtäsuuri. Tulosta voi kysyä lähdöstä OUT. Mikäli esiintyy ylivuoto, ENO = 0. Mikäli tulo ei ole liukuluku, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
	<p>Operaatio seuraavaksi pienemmän kokonaisluvun muodostus liukuluvusta lukee tuloparametrin IN sisällön liukulukuna ja muuntaa tämän kokonaisluvuksi (32 bittiä) (esimerkki: +1,5 -> +1; -1,5 -> -2). Tulos on suurin kokonaisluku, joka on annettua liukulukua pienempi/yhtäsuuri. Tulosta voi kysyä lähdöstä OUT. Mikäli esiintyy ylivuoto, ENO = 0. Mikäli tulo ei ole liukuluku, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>

6.3 Esimerkki

Muunto suoritetaan, jos E 0.0 = 1. Merkkerisanan MW10 sisältö luetaan kolminumeroisena BCD-lukuna ja muunnetaan kokonaisluvuksi (16 bittiä). Tulos tallennetaan merkkerisanaan MW12. Mikäli muunto suoritetaan, A 4.0 = 1 (ENO = EN) (kuva51). /2/



Kuva 51 Esimerkki BCD-luvun muunnosta kokonaisluvuksi (16 bittiä)

7 VERTAILIJAT(Comparator)

7.1 Kokonaislukujen vertailu (16 bittiä)

Operaatio kokonaislukujen vertailu (16 bittiä) suorittaa 16 bitin kokonaisluvun vertailuoperaation. Tuloja IN1 ja IN2 verrataan dialogikentässä valitun vertailulajin mukaan. Vertailulajit on lueteltu taulukossa 8. Kokonaislukujen vertailijoiden (16 bittiä) symbolit kuvassa 53. /2/

Taulukko 8 Tuloja IN1 ja IN2 verrataan seuraavien vertailulajien mukaisesti. /2/

Vertailulaji	Merkki
IN1 on yhtäsuuri kuin IN2.	==
IN1 on erisuuri kuin IN2.	<>
IN1 on suurempi kuin IN2.	>
IN1 on pienempi kuin IN2.	<
IN1 on suurempi tai yhtäsuuri kuin IN2.	>=
IN1 on pienempi tai yhtäsuuri kuin IN2.	<=

Mikäli vertailun tulos on ”tosi”, on operaation tulo ”1”, muuten ”0”.

Vertailutuloksella ei ole negatiota, koska tämä voidaan saavuttaa kulloinkin käänteisellä vertailuoperaatiolla. /2/

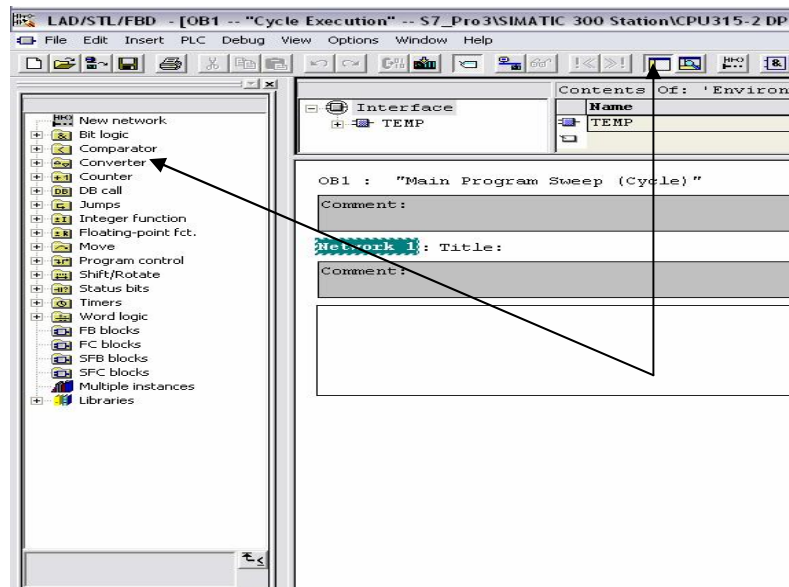
Käytettävissä ovat seuraavat vertailuoperaatiot:

1. CMP ? I Kokonaislukujen vertailu (16 bittiä)
2. CMP ? D Kokonaislukujen vertailu (32 bittiä)
3. CMP ? R Liukulukujen vertailu. /2/

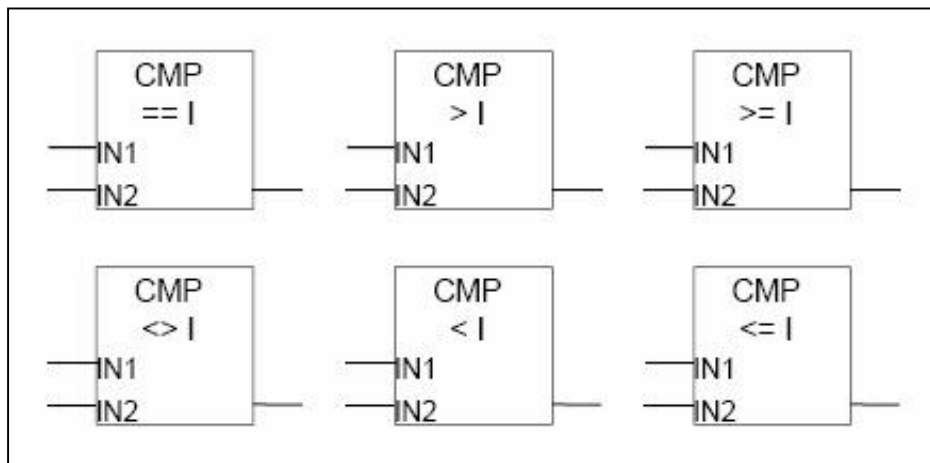
7.2 Operaation valinta Simatic Manager -ohjelmassa

Käynnistä Simatic Manager –ohjelma Wizard -ohjeiden mukaisesti.

Käskyt (kuvat 53 ,55 ja 56) löytyvät Simatic- ohjelmasta kohdasta Program Elements→Comparator (kuva 52).



Kuva 52 Comparator -operaatioiden valinta



Kuva 53 Kokonaislukujen vertailijoiden (16 bittiä) symbolit. /2/

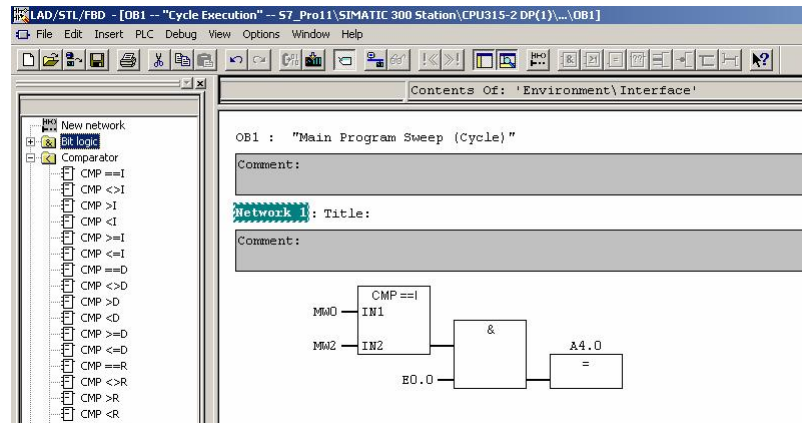
Operaatio **kokonaislukujen vertailu (16 bittiä)** suorittaa vertailuoperaation 16 bitin kokonaisluvun perusteella. Tulosta IN1 ja IN2 verrataan valitun vertailulajin mukaan. /2/

7.3 Esimerkki

A 4.0 asetetaan, jos (kuva 54):

MW0 = MW2

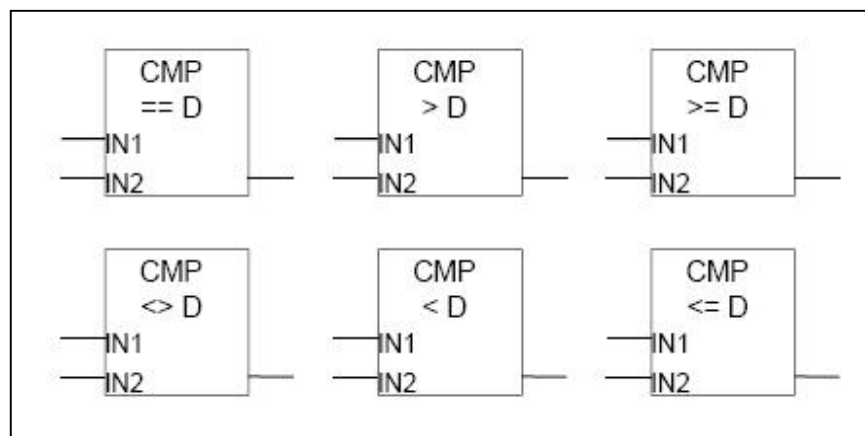
JA tulon E 0.0 signaalitila on "1". /2/



Kuva 54 Esimerkki vertailija operaatiosta (16 bittiä)

7.4 Kokonaislukujen vertailu (32 bittiä)

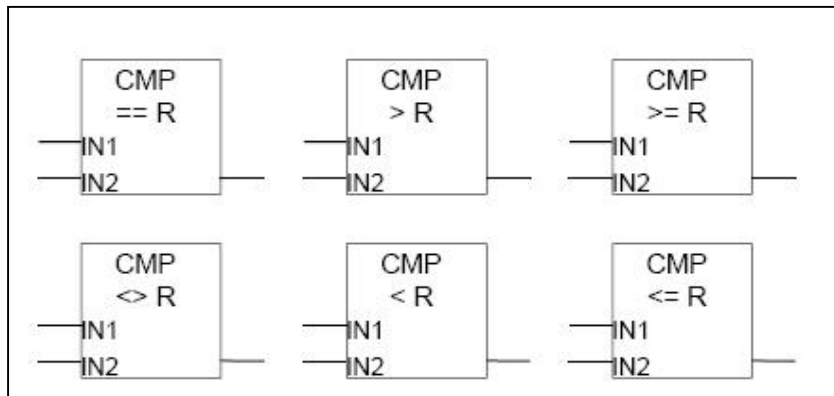
Operaatio **kokonaislukujen vertailu (32 bittiä)** suorittaa vertailuoperaation 32 bitin kokonaisluvun perusteella. Tulosta IN1 ja IN2 verrataan valitun vertailulajin mukaan. Vertailijoiden symbolit kuvassa 55. /2/



Kuva 55 Kokonaislukujen vertailijoiden (32 bittiä) symbolit. /2/

7.5 Liukulukujen vertailu

Operaatio liukulukujen vertailu suorittaa vertailuoperaation reaalityökaluilla
Tuloja IN1 ja IN2 verrataan valitun vertailulajin mukaan. Vertailijoiden
symbolit ovat kuvassa 56. /2/



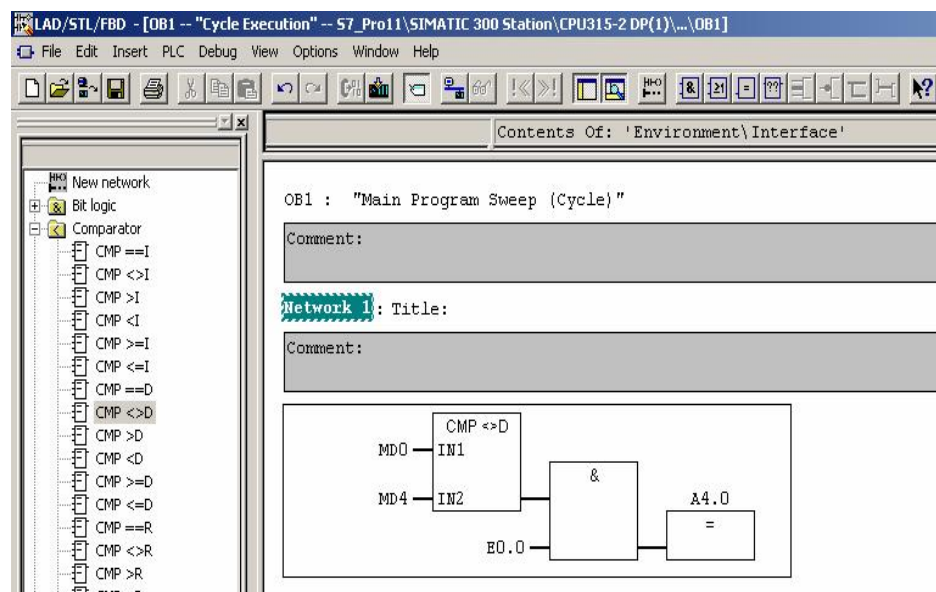
Kuva 56 Liukulukujen vertailijoiden symbolit /2/

7.6 Esimerkki

A 4.0 asetetaan, jos (kuva 57):

MD0 on erisuuri kuin MD4

JA tulon E 0.0 signaalitila on "1". /2/



Kuva 57 Esimerkki vertailijaoperaatiosta (32 bittiä)

8 PERUSLASKENTAOPERAATIOT

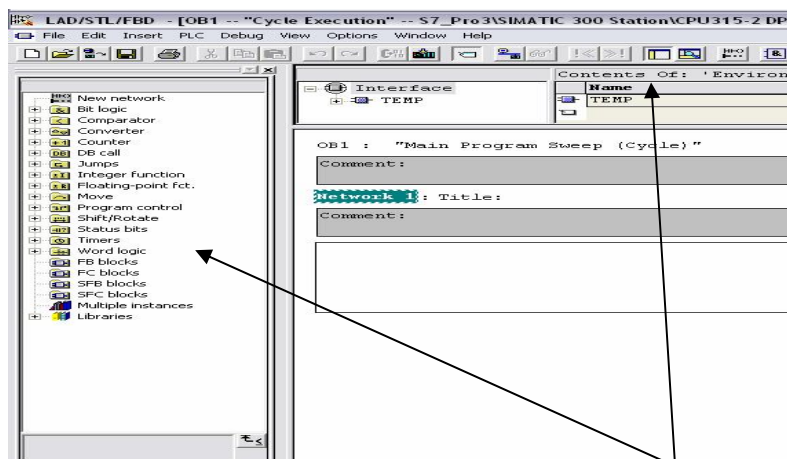
8.1 Yleiskuva kokonaislukutoiminnoista

Kokonaislukutoiminnoilla seuraavat operaatiot voidaan suorittaa kahdella kokonaisluvulla (16 bittiä, 32 bittiä):

1. ADD_I kokonaislukujen yhteenlasku (16 bittiä)
2. SUB_I kokonaislukujen vähennyslasku (16 bittiä)
3. MUL_I kokonaislukujen kertolasku (16 bittiä)
4. DIV_I kokonaislukujen jakolasku (16 bittiä)
5. ADD_DI kokonaislukujen yhteenlasku (32 bittiä)
6. SUB_DI kokonaislukujen vähennyslasku (32 bittiä)
7. MUL_DI kokonaislukujen kertolasku (32 bittiä)
8. DIV_DI kokonaislukujen jakolasku (32 bittiä)
9. MOD_DI jakojäännös (32 bittiä). /2/

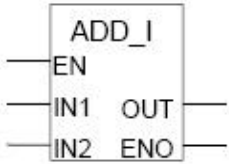
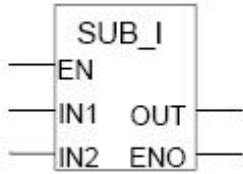
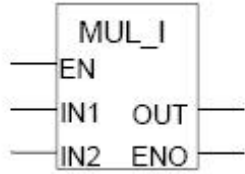
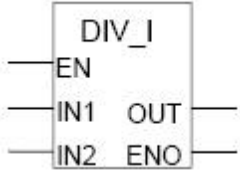
8.2 Operaation valinta Simatic Manager -ohjelmassa

Käynnistetään Simatic Manager -ohjelma Wizard -ohjeiden mukaisesti. Käskyt (Taulukot 9-11) löytyvät Simatic- ohjelmasta kohdasta Program Elements → Integer Function (kuva 58).

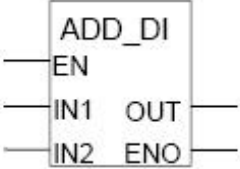
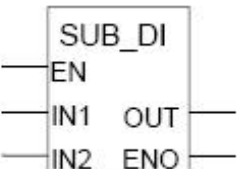
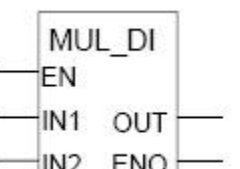


Kuva 58 Integer Function –valinta

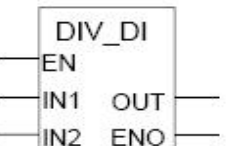
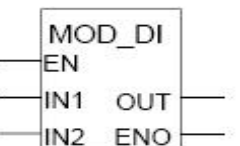
Taulukko 9 Käskyjen kuvaus /2/

Käsky	Kuvaus
<p>Yhteenlasku (16bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen yhteenlasku (16 bittiä). Tämä operaatio laskee yhteen tulot IN1 ja IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (16 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Vähennyslasku (16bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen vähennyslasku (16 bittiä). Tämä operaatio vähentää tulosta IN1 tulon IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (16 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Kertolasku (16bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen kertolasku (16 bittiä). Tämä operaatio kertoo tulon IN1 tulolla IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (16 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Jakolasku (16bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen jakolasku (16 bittiä). Tämä operaatio jakaa tulon IN1 tulolla IN2. Tämän jakolaskun osamäärää (kokonaislukuosuutta) voidaan kysyä lähdöstä OUT. Jakojäännöstä ei voi kysyä. Osamäärän ollessa kokonaislukujen (16 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>

Taulukko 10 Käskyjen kuvaus /2/

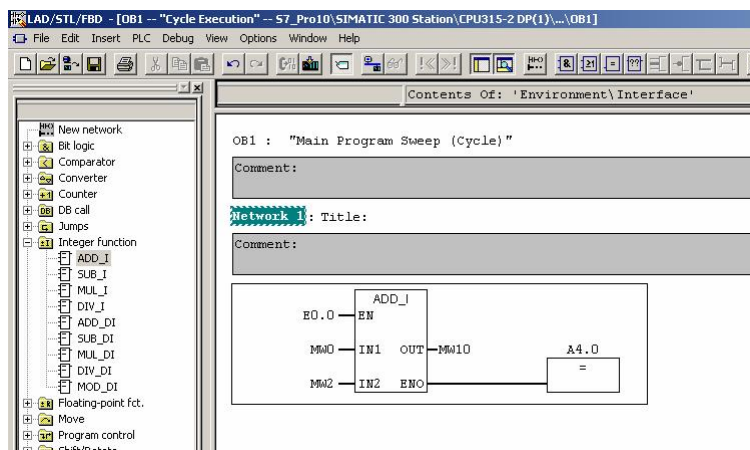
Käsky	Kuvaus
<p>Yhteenlasku (32 bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen yhteenlasku (32 bittiä). Tämä operaatio laskee yhteen tulot IN1 ja IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (32 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Vähennyslasku (32 bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen vähennyslasku (32 bittiä). Tämä operaatio vähentää tulosta IN1 tulon IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (32 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Kertolasku (32 bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen kertolasku (32 bittiä). Tämä operaatio kertoo tulon IN1 tulolla IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaislukujen (32 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>

Taulukko 11 Käskyjen kuvaus /2/

Käsky	Kuvaus
<p>Jakolasku (32 bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation kokonaislukujen jakolasku (32 bittiä). Tämä operaatio jakaa tulon IN1 tulolla IN2. Tämän jakolaskun osamäärää (kokonaislukuosuutta) voidaan kysyä lähdöstä OUT. Operaation kokonaislukujen jakolasku (32 bittiä) tallentaa osamäärän yksinkertaisena 32 bitin arvona DINT-muodossa eikä siitä jää jakojäännöstä. Osamäärän ollessa kokonaislukujen (32 bittiä) sallitun alueen ulkopuolella, on OV-bitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>
<p>Jakojäännös (32 bit.)</p> 	<p>Vapautustulon (EN) signaalitila "1" aktivoi operaation jakojäännös (32 bittiä). Tämä operaatio jakaa tulon IN1 tulolla IN2. Tulosta voi kysyä lähdöstä OUT. Tuloksen ollessa kokonaisluvulle (32 bittiä) sallitun alueen ulkopuolella, on OVbitin ja OS-bitin arvo "1" ja ENO:n arvo "0".</p>

8.3 Esimerkki 1

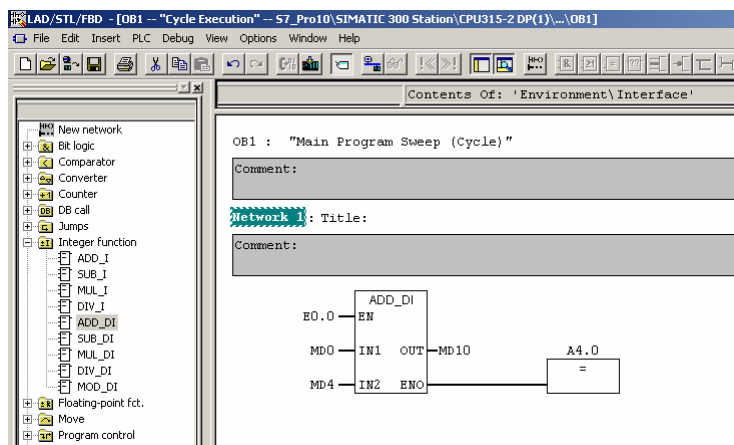
Kehys ADD_I aktivoidaan, jos E 0.0 = 1. Yhteenlaskun MW0 + MW2 tulos tallennetaan merkkierisanaan MW10. Mikäli tulos on kokonaisluvulle (16 bittiä) sallitun alueen ulkopuolella tai tulon E 0.0 signaalitila = 0, osoitetaan lähdölle A 4.0 signaali "0" eikä operaatiota suoriteta (Kuva 59). /2/



Kuva 59 Käsken ADD_I esimerkki

8.4 Esimerkki 2

Kehys ADD_DI aktivoidaan, jos E 0.0 = 1. Yhteenlaskun MD0 + MD4 tulos tallennetaan merkkierikaksoissanaan MD10. Mikäli tulos on kokonaisluvulle (32 bittiä) sallitun alueen ulkopuolella tai tulon E 0.0 signaalitila = 0, osoitetaan lähdölle A 4.0 signaali "0" eikä operaatiota suoriteta (kuva 60). /2/



Kuva 60 Käsken ADD_DI esimerkki

9 SIIRTOKÄSKY (MOVE)

9.1 Kuvaus

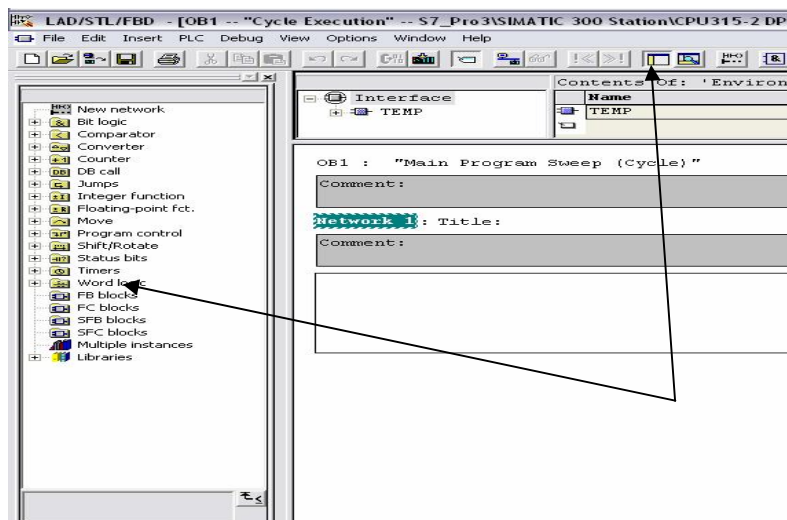
Arvon siirto-operaatiolla voidaan varata muuttujia, joilla on erikoisarvoja. Tulossa IN annettu arvo kopioidaan lähdössä OUT annettuun operandiin. ENO:n signaalitila on sama kuin EN:n. /2/

Arvon siirto-operaatio pystyy kopioimaan MOVE -kehyksellä kaikki 8, 16 tai 32 bitin pituiset tiedostotyytit. Käyttäjämääritteiset tiedostotyytit kuten kentät ja rakenteet on kopioitava systeemitominnalla SFC 20 "BLKMOV" (katso referenssikäsikirja /2/). /2/

Master Control Relay (MCR) vaikuttaa operaatioon *arvon siirto*. Lähempää tietoa MCR:n toimintatavasta löytyy referenssikäsikirjasta./2/

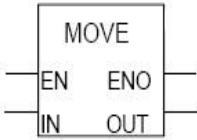
9.2 Käskeyjen valinta Simatic Manager –ohjelmassa

Käynnistetään Simatic Manager –ohjelma Wizard -ohjeiden mukaisesti. Käsky (Taulukossa 12) löytyy Simatic- ohjelmasta kohdasta Program Elements→ Move (kuva 61).



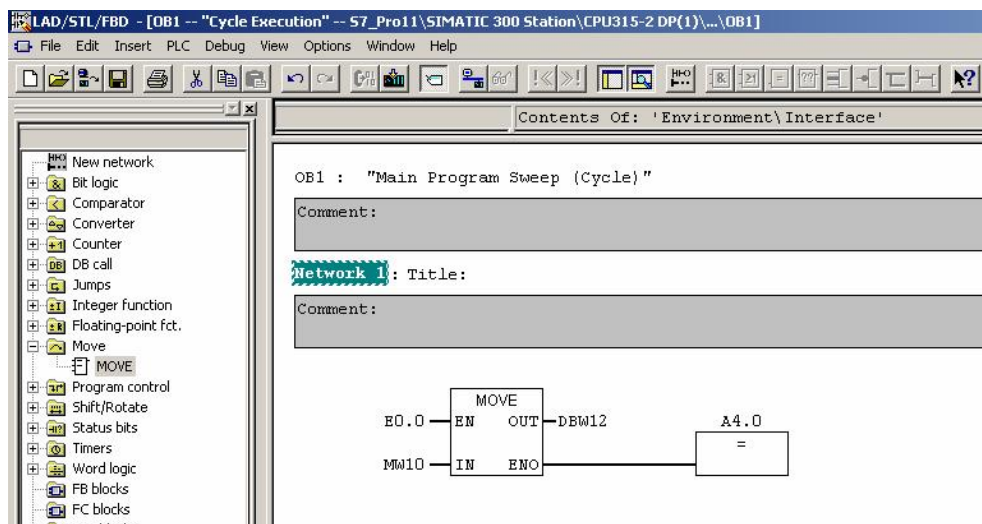
Kuva 61 Move -käskyn valinta

Taulukko 12 Siirtokäskyn kuvaus /2/

Käsky	Kuvaus
	<p>Arvon siirto-operaatiolla voidaan varata muuttujia, joilla on erikoisarvoja. Tulossa IN annettu arvo kopioidaan lähdessä OUT annettuun operandiin. ENO:n signaalitila on sama kuin EN:n. Operaatio arvon siirto pystyy kopiaamaan MOVE-kehyksellä kaikki 8:n, 16:n tai 32:n bitin pituiset perustiedostotyypit. Käyttäjämääritteiset tiedostotyypit kuten kentät ja rakenteet on kopioitava systeemitominnalla SFC 20 "BLKMOV". Master Control Relay vaikutta operaatioon arvon siirto.</p>

9.3 Esimerkki

Operaatio suoritetaan, jos $E\ 0.0 = 1$. MW10:n sisältö kopioidaan avatun tiedostoyksikön/DB datasanaan 12. Mikäli operaatio suoritetaan, $A\ 4.0 = 1$ (kuva 62). /2/



Kuva 62 Move -käskyn esimerkki

10 YKSIKÖIDEN PARAMETROINTI

10.1 Yleistä

Koska logiikkalaitteisto voi sisältää tyypiltään erilaisen keskusyksikön ja erilaisia I/O-yksiköitä, on ohjelman toimiakseen tiedettävä laitteiston tarkka kokoonpano. Parametrointi on erilaisten asetusten tekemistä laitteiston yksiköille. /2/

10.2 Parametrointi

Parametrointi tapahtuu valitsemalla hiirellä konfigurointitaulukon riviltä laite, jonka arvoja halutaan muuttaa. Parametroinnilla voidaan vaihtaa esimerkiksi keskusyksikön ohjelman kiertoaikaa tai määritellä I/O-yksiköiden tulojen ja lähtöjen osoitteistot (osoitteiston määrittäminen ainoastaan keskusyksikkömallissa 315-2 DP). /2/

Kun kutsutaan yksikköä, voidaan sille antaa parametreja ja arvoja, joilla sen tulee toimia. Nämä arvot ja parametrit ovat nimeltään yksikköparametreja. Parametrien antaminen merkitsee, että yksikkö tarvitsee näitä käskyosuudellaan. Täten voidaan muodostaa universaaleja useampia kertoja käytettäviä yksiköitä. Kun yksikköä kutsutaan, tulee/voidaan sille antaa arvot tai osoitteet. Jotta näitä osoitteita voidaan siirtää yksiköiden välillä, tulee yksiköille määrittää parametrit. Selvitys tapahtuu muuttujataulukossa (kuva 63), jossa annetaan paikallisdatat ja datatyypit. Paikallisdatoilla (L) on seuraavat tyypit. Aktuaaliparametreja ei ole staattisilla eikä hetkellisillä (temp) parametreilla. /2/

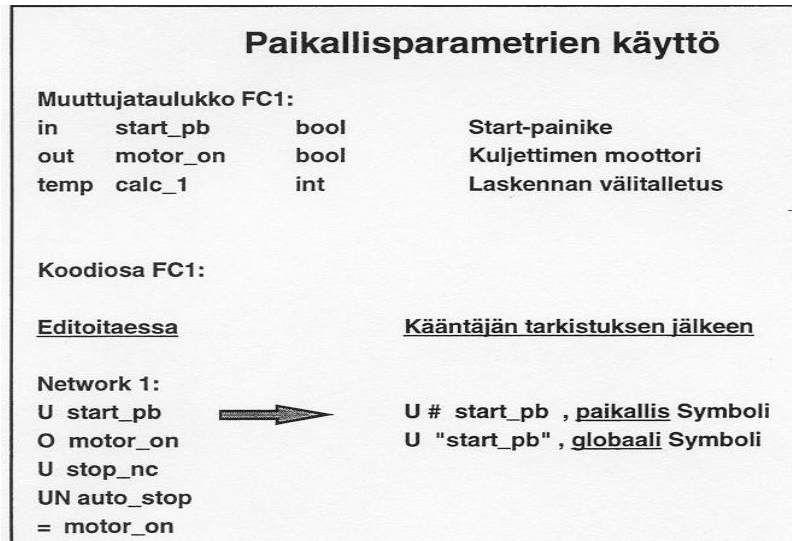
10.3 Parametri

Määrittää tuloarvot (vain luku), lähtöarvot (vain kirjoitus) ja tulo/lähtöarvot (luku- ja kirjoitus), joilla yksikkö liittyy muuhun ohjelmaan.

Parametrityypit ovat:

1. TYPE tyyppi määrittelee, miten parametria käytetään yksikössä
2. IN tuloparametri, jota luetaan
3. OUT lähtöparametri, jota ohjataan
4. IN OUT luku tai kirjoitus mahdollinen
5. STAT paikallinen muuttuja, jonka arvo talletetaan oheis-DB:lle
6. TEMP hetkellinen muuttuja, jonka arvoa ei enää talleteta yksikö jättämisen jälkeen.
7. NAME muuttujan symbolinen nimi, jota yksikön käskyosa käyttää
8. DATATYP datatyyppi jokaiselle muuttujalle. Esim: BOOL (Boole), INT (Integer)
9. Initial value vaihtoehtoinen kenttä alkuarvon tai käynnistysarvon annolle.
10. Comments vaihtoehtoinen kenttä muuttujan kommentoimiseksi. /2/

Kuvassa 63 on selvitetty paikallis parametrien käyttöä.



Kuva 63 Paikallisparametrien käyttö /2/

10.4 Staattiset arvot

Vain FB määrittelee muuttujan (stat"), joka talletetaan oheis DB:lle.

Staattiset arvot (stat) ovat olemassa vain FB-yksiköillä. (FB:t joilla on oheis-DB). /2/

10.5 Hetkelliset arvot

Hetkelliset arvot määrittelee muuttujan ("temp"), joka tallentuu paikallisdataan (L). Tämä data on voimassa vain niin kauan kun yksikköä läpikäydään. Kun yksikkö on läpikäyty, niin L-data saa uudet sisällöt muusta yksiköstä ja vanhat arvot katoavat. Kaikki yksikön muuttujaparametreina merkityt arvot ovat nimeltään muodollisia parametreja, joihin kytketään yksikköä kutsuttaessa todellinen aktuaaliparametri. /2/

Parametrien muuttujataulukko (kuva 64) sisältää määritteet parametrin nimen, muodon, nimen datatyypin, alkuarvon ja kommentin. Jokaiselle riville tulee oma muuttuja. ARRAY- tyyppillä voidaan muodostaa useampia rivejä. Toiminta on FBD / FUP, LAD / KOP ja STL / AWL muodolle samanlainen. /2/

Paikallissymbolit ja paikallis (L) -muisti

Osoite
↓
Nimi
↓
Alkuarvo
↓

Address	Decl.	Name	Type	Initial Value	Comment
0.0	in	on	BOOL	FALSE	valo päällä
0.1	in	moottori	BOOL	FALSE	moottoritieto
0.2	in	kaynnistys	BOOL	FALSE	kaynnistystieto
2.0	out	arvo	INT	0	moottoriarvo
4.0	in-out	siirto	BYTE	B#16#0	asetuspiste
		temp			

↑
Parametrityyppi
↑
Datatyyppi
↑
Kommentti

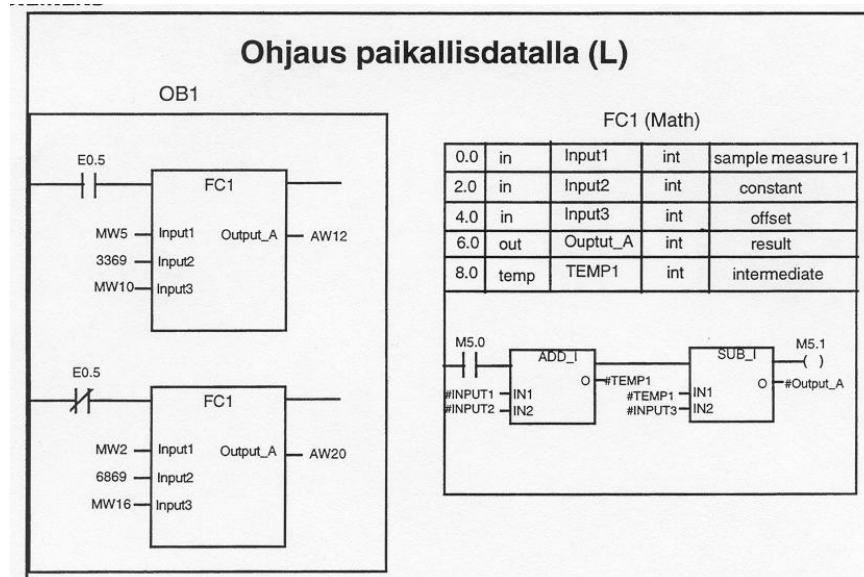
Kuva 64 Parametrien muuttujataulukko /2/

10.6 Paikallisosoitteet

Jokainen STEP 7 muuttuja varaa tietyn muistialueen. Paikallisosoitte näkyy parametrin kirjoittamisen jälkeen. Osoitteena muistissa on osoiteryhmä (BYTE.BIT) tai datatavunumero. Paikallisosoitteen käsittely voi tapahtua

suoraan absoluuttiosoitteella, on kuitenkin suositeltavaa käyttää ohjelmassa symbolinimeä. /2/

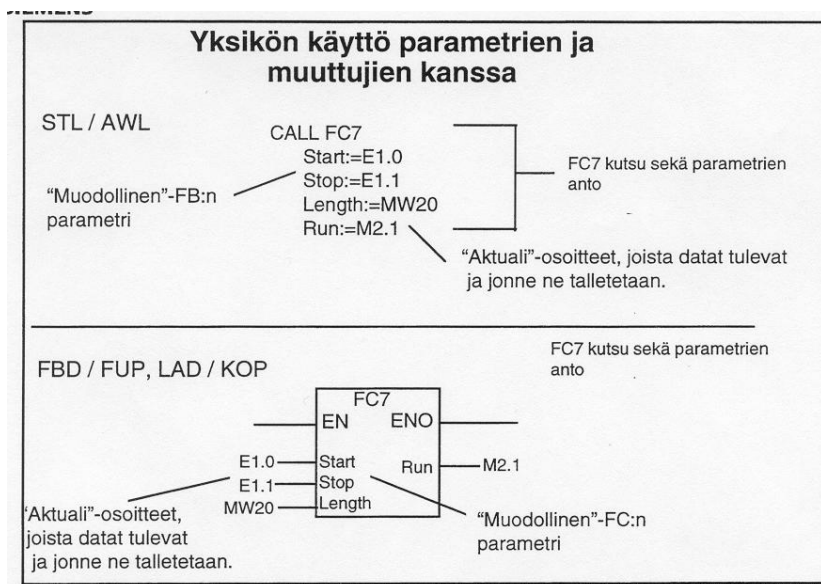
Yksikköä kutsuttaessa käyttöjärjestelmä varaa yksikölle muistialueen, jota nimitetään L-Stackiksi (rekisteri). Tämä hetkellinen muisti on tehokas väline väliarvojen tallettamiseksi, jotka eivät ole enää tarpeellisia yksikön jättämisen jälkeen. Yksikön parametrin selvitysosassa muodostettuja L-muistin sisällä olevia nimiä ja tyyppejä käytetään yksikön sisällä olevassa ohjelmassa. Jotta näitä paikallisia yksikköparametreja voidaan käyttää käskyosassa, puhutellaan niitä selvitysosassa olevalla nimellä. Nimi tulee antaa oikein suurin ja pienin kirjaimin. Virheellisessä annossa editor ilmoittaa virhettä. Oheiset käskyesimerkit esittävät käytettyä symbolinimen käyttöä. Nämä symbolit ovat käytössä vain tässä yksikössä. Mikäli halutaan käyttää symbolinimeä, jota käsitellään useammassa yksikössä, tulee tämä nimi muodostaa globaalissa symboliparametritaulukossa. Ohjelmaeditori näyttää omilla merkinnöillä, kumpaan ryhmään symboliosoite kuuluu. /2/



Kuva 65 Esimerkki FC1:n kutsusta /2/

Käytettäessä paikallismuistia FC:N tai FB:n sisällä tulee yksikön selvitysosassa antaa parametrin nimi ja datatyyppi. Kun FC / FB yksikköä kutsutaan, annetaan aktuaaliparametrit näille paikallisdatoille. Kuvassa 65 olevassa esimerkissä suoritetaan FC1:n ehdollinen kutsu tuloilla EO.5. FC1:lle annetaan eri arvoja, ja lähtöarvo annetaan OB1:lle takaisin. /2/

Kuvassa 66 on selvitetty yksikön käyttöä parametrien ja muuttujien kanssa.



Kuva 66 Yksikön käyttö parametrien ja muuttujien kanssa /2/

11 Analoginen ohjelma

11.1 Automaatiolaitteen osoitteet

Korttipaikka 1

Virtalähdeyksikkö on useimmiten ensimmäisellä paikalla. Tällä paikalla ei ole korttipaikkaosoitetta. /2/

Korttipaikka 2

Keskusyksikkö (CPU). Sen tulee sijaita virtalähteen oikealla puolella. Tällä paikalla ei ole korttipaikkaosoitetta. /2/

Korttipaikka 3:

Liitäntäyksikkö (IM). Laajennuskehikoiden (lisäkiskojen) liittämiseksi. IM-yksiköillä ei ole korttipaikkaosoitetta. Myös silloin kun korttia ei ole käytössä, se varaa oman paikanumeronsa. Korttipaikka 3 on varattu CPU-rivillä IM-yksikköä varten. /2/

Paikat 4-11

Signaaliyksiköt. Paikka 4 on varsinainen ensimmäinen tulo / lähtöyksikkö (kuva 2). /2/

11.2 Analogiaosoitteet

Korttipaikan mukaan muodostuvat analogiosoitteet alkavat numerosta 256, mikäli analogiakortti on korttipaikassa 4 (kuva 68). Jokainen korttipaikka varaa 16 tavua osoitealueesta. Jokainen kanava käyttää kaksi tavua, joten osoitteen

tulee alkaa parillisella luvulla. 314 IFM laitteessa on integroidut 4 analogiatuloa ja yksi analogialähtö, joiden osoitteet alkavat tulosanasta PEW128. /2/

S7-300 ja S7-400 laitteilla ei ole, kuten digitaalipuolella, sykleittäin päivittyvää prosessikuvaa, vaan kortteja puhutellaan suoraan. Analogitulon parametri on PEW ja lähdön PAW. Koska osoitealue alkaa numerosta 256, on (Kuva 67) ensimmäisellä korttipaikalla olevan analogikortin osoite PEW 256. Viimeinen osoite on 766. /2/

SIEMENS

Tulo / lähtöyksiköt - Analogi

	Slot #	3	4	5	6	7	8	9	10	11
Rack 3	Virta-lähde	IM 361	640	656	672	688	704	720	736	752
			-	-	-	-	-	-	-	-
Rack 2	Virta-lähde	IM 361	512	528	544	560	576	592	608	624
			-	-	-	-	-	-	-	-
Rack 1	Virta-lähde	IM 361	384	400	416	432	448	464	480	496
			-	-	-	-	-	-	-	-
Rack 0	CPU ja Virta-lähde	IM 360	256	272	288	304	320	336	352	368
			-	-	-	-	-	-	-	-
			270	286	302	318	334	350	366	382

Kuva 67 Korttipaikkojen osoitteet /2/

← → (0) UR

Slot	Module	Order number	Firmware	MPI address	I address	Q address
1						
2	CPU 314 IFM	6ES7 314-5AE03-0AB0	V1.2	2	124...135	124...129
3						
4	AI2x12Bit	6ES7 331-7KB81-0AB0			256...259	
5	AI4x0/4 to 20mA, Ex	6ES7 331-7RD00-0AB0			272...279	
6	AI8x12Bit	6ES7 331-7KF02-0AB0			288...303	
7						
8						
9						

Kuva 68 Korttipaikan mukaan muodostuvat analogiaosoitteet /1/

11.3 Analogiaosoitteet S7-400

Myös S7 -400 laitteella on oletusosoitteet. Alkuosoite lasketaan:

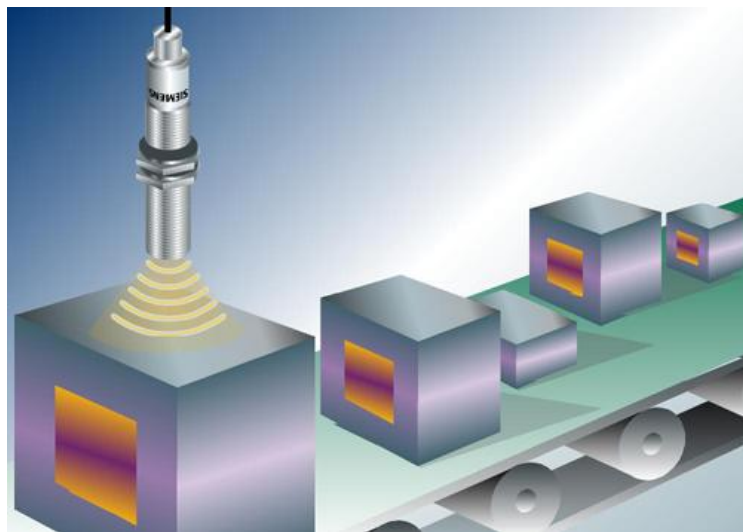
Analogialkuosoite = (fysikaalinen paikkanumero -1) x 64 + 512

Esimerkki:

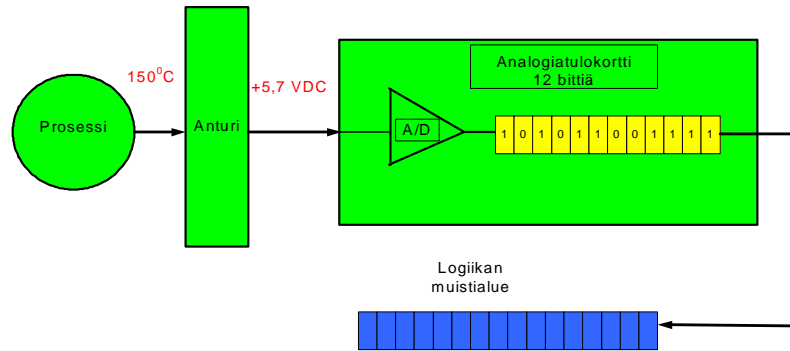
Alkuosoite paikalla 4 on 704. /1/

11.4 Liukuhihna esimerkki

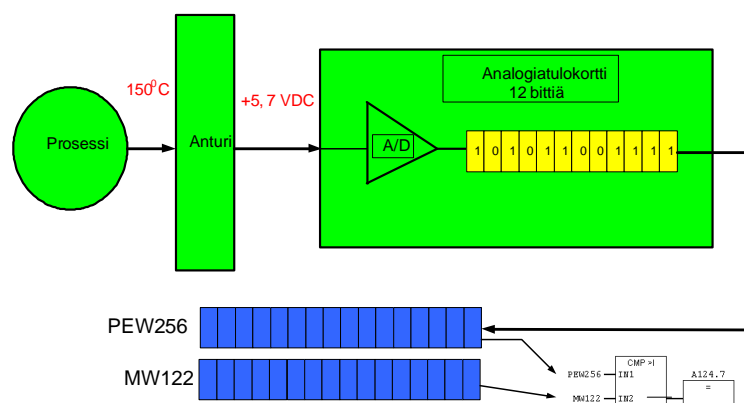
Kuvissa 69-73 on kuvattu teollisuudessa käytössä oleva järjestelmä. Anturi mittaa hihnalla kulkevien kappaleiden lämpötilaa. Raja-arvona on 150 °C.



Kuva 69 Kuva prosessista



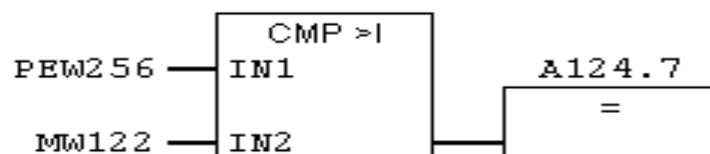
Kuva 70 Anturi antaa tietoa logiikan muistille



Kuva 71 Järjestelmään lisätty vertailijaoperaatio

Yleisesti käytetyt analogiaviestit
4–20 mA 0 to +1 volts DC 0 to +5 volts DC 0 to +10 volts DC 1 to +5 volts DC ± 5 volts DC ± 10 volts DC

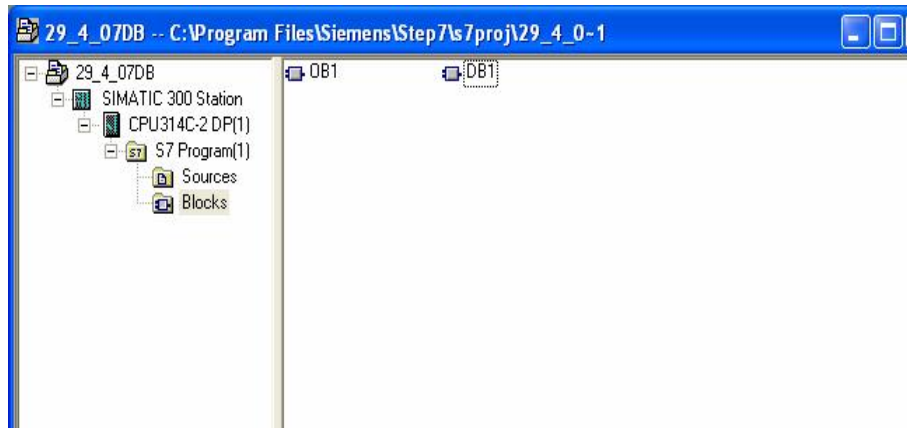
Kuva 72 Yleisimmät analogiaviestit



Kuva 73 Vertailijaoperaatio

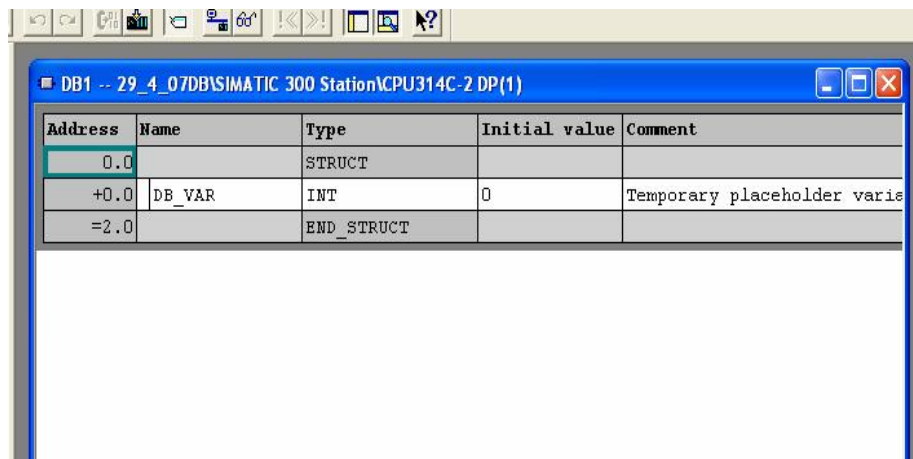
12 DB -yksikön alustus

Käynnistetään Simatic Manager –ohjelma Wizard ohjeiden mukaisesti. Näin päästään kuvan 74 ikkunaan. /1/



Kuva 74 DB yksikön alustus /1/

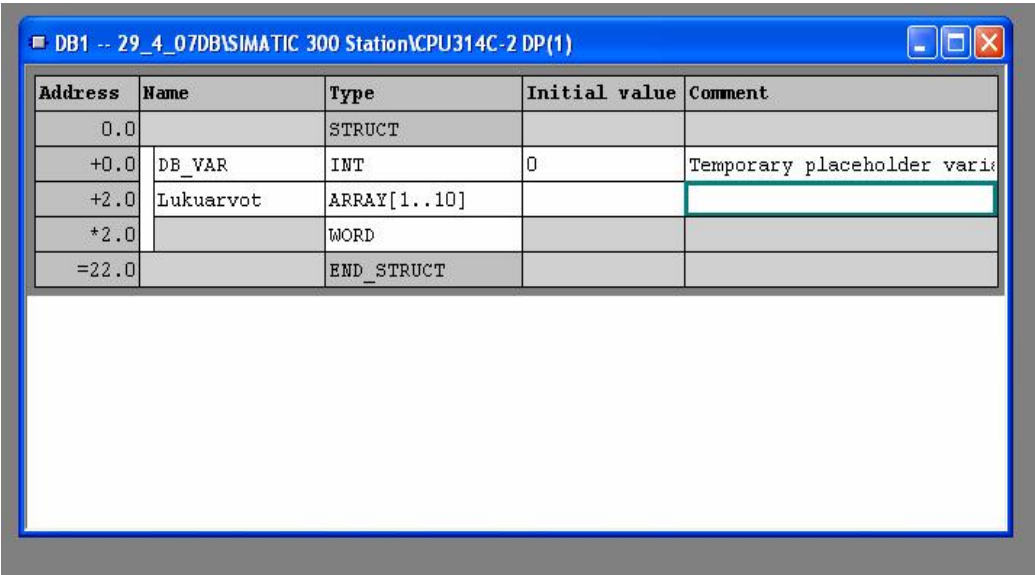
DB1-yksikköön alustetaan tässä esimerkissä kymmenen muistipaikkaa WORD-muotoiselle tiedolle. DB-yksikön alustus suoritetaan yksikön otsikkotietoihin. Otsikkotiedot avautuvat kuvan 75 mukaisessa muodossa. /1/



Kuva 75 Otsikkotiedot /1/

Name-sarakkeeseen annetaan tietoa kuvaava nimi ”*Lukuarvo*” (kuva 76). Type-sarakkeessa määritellään tarvittava tietojen määrä ”*ARRAY[1..10]*”). Tässä esimerkissä luodaan kymmenen riviä ”*WORD*”-tyyppistä muistipaikkaa.

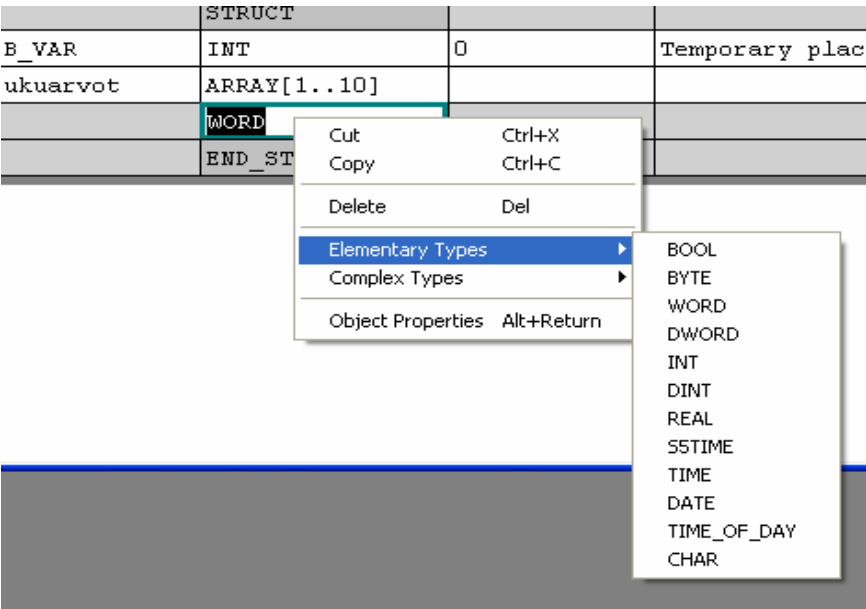
Tiedot voivat olla myös muita kuin WORD muotoisia muistipaikkoja, kuten kuvasta 77 nähdään. /1/



The screenshot shows a window titled "DB1 -- 29_4_07DBSIMATIC 300 Station\CPU314C-2 DP(1)". It contains a table with the following data:

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	DB_VAR	INT	0	Temporary placeholder variable
+2.0	Lukuarvot	ARRAY[1..10]		
+2.0		WORD		
=22.0		END_STRUCT		

Kuva 76 Nimeäminen /1/



The screenshot shows a table with the following data:

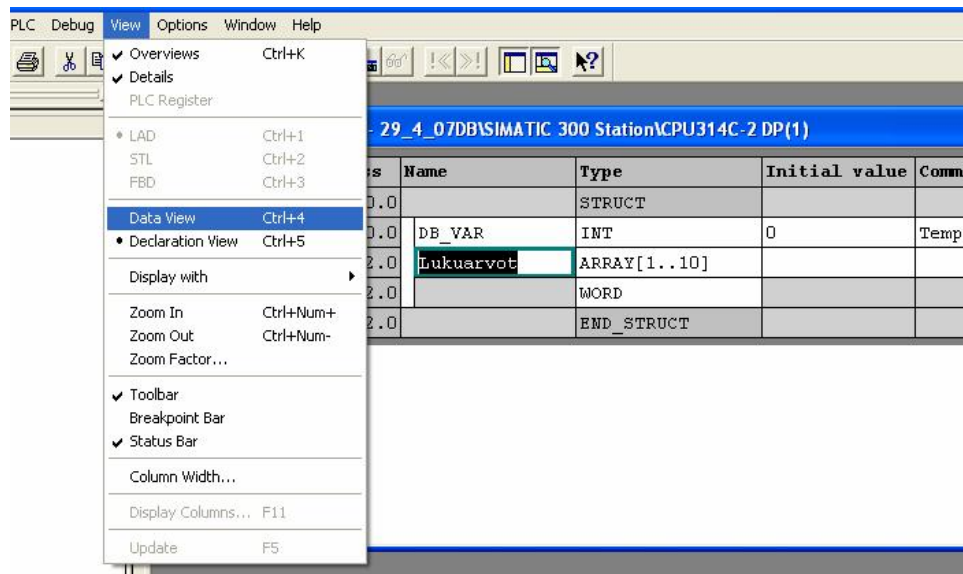
	STRUCT		
B_VAR	INT	0	Temporary plac
ukuarvot	ARRAY[1..10]		
	WORD		
	END_ST		

A context menu is open over the 'WORD' entry, showing the following options:

- Cut (Ctrl+X)
- Copy (Ctrl+C)
- Delete (Del)
- Elementary Types
 - BOOL
 - BYTE
 - WORD
 - DWORD
 - INT
 - DINT
 - REAL
 - SSTIME
 - TIME
 - DATE
 - TIME_OF_DAY
 - CHAR
- Complex Types
- Object Properties (Alt+Return)

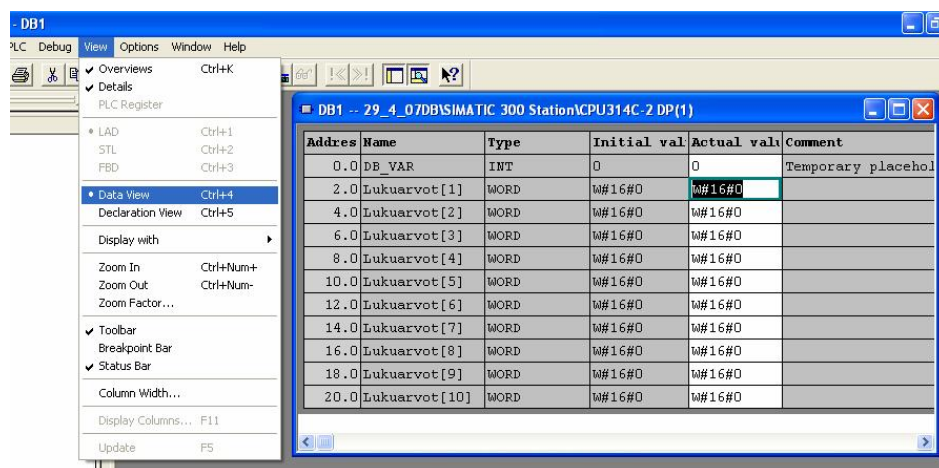
Kuva 77 Tiedot voivat olla myös muita kuin WORD muotoisia muistipaikkoja /1/

Valitsemalla verhovalikosta ”View ”-> ”Data View” (kuvas 78), nähdään Word-tiedot riveittäin. /1/



Kuva 78 WORD –tiedot /1/

Avautuvasta taulukosta (kuva 79) nähdään WORD -muistipaikkojen osoitteet. Osoitealue on tässä esimerkissä tavuosoite 2:sta tavuosoitteeseen 20, huom! yksi ”Word” sisältää 2 tavua. /1/



Kuva 79 WORD -muistipaikkojen osoitteet /1/

13 TULOKSET

Tämän tutkintotyön tuloksena on kappaleessa 2 esitetty ohjekirja. Ohjekirja on 74-sivuinen. Ohjekirja sisältää pääasiassa käyttöohjeet kahdelle logiikkaväylälle sekä antaa tietoa niiden soveltamisesta käytännössä. Se antaa myös jonkin verran perustietoa ohjelmoitavista logiikoista sekä muista automaatioon liittyvistä aiheista.

14 TULOSTEN TARKASTELU

Käyttökelpoisuutensa kannalta tarkasteltuna ohjekirja sopii erittäin hyvin tehtäväänsä. Se on selkeä, tiivis ja helposti ymmärrettävä kokonaisuus aiheestaan. Vaikeita asioita on kuvattu esimerkkien ja kuvien avulla, mikä edesauttaa uuden asian oppimista. Mahdolliset virheet ja puutteet on pyritty minimoimaan kokoamalla käyttöohjeet vaihe vaiheelta ohjelman etenemisen aikana.

Ohjekirjan kirjoittaminen onnistui suurelta osin ilman ongelmia. Tekstimäärän kasvaessa ohjeistuksen kokonaisuuden hahmottaminen ja ratkaisujen tekeminen helpottuivat. Muutamaa poikkeusta lukuun ottamatta ohjekirjan asiakohdat valmistuivat yhdellä kertaa.

Ohjekirja on täyttänyt sille alussa annetut tavoitteet. Se täydentää aiemmin opittuja tietoja sekä opettaa täysin uusia asioita. Koska ohjeistus on tiivis tietopaketti Simatic Manager S7:n jatkokurssista, se edellyttää käyttäjältään peruskurssin suorittamista.

Sivumäärän rajallisuuden vuoksi aivan kaikkia aihepiirejä ei ole voitu esitellä kaiken kattavasti. Käytännössä yleisimmät aihepiirit on tarkasteltu ohjekirjassa tarkemmin.

LÄHTEET

Painetut lähteet

- 1 Mäkelä, Seppo, Simatic Step S7 Ohjelmointiohje. Tampereen ammattikorkeakoulu. Koneosasto. 2005.

Painamattomat lähteet

- 2 Yksiköiden ohjelmointi toimintakaaviomuodossa FBD/FUP S7-300/400. Referenssikäsikirja. Siemens Simatic. Versio 11/2002 FIN. 212 s.
- 3 Suominen, Sami – Ahava, Veikko, Asi- väyläohje. Ohjelmoitavat logiikat. Tampereen ammattikorkeakoulu. Esitelmä 2006.